

GAME MANUAL

MOONING
NET

COMMAND MODERN OPERATIONS

ON

JOHN S. MCCAIN (DDG-56)
LEIGH BURKE-CLASS

AGAN (CVN-76)

BANDITS
J-20 SQUADRON
PLAAF

2 F/A-18E SUPER HORNET
RAAF



Contents

Preface	1
Historical Introduction	1
What is Command?	3
1.0 Installation	4
1.1 System Requirements	4
1.2 Support	4
1.3 Notes for Multitaskers and Returning Players	4
2.0 Introduction to COMMAND	6
2.1 Important Terms	7
2.2 Fundamentals	8
2.2.1 Starting COMMAND	8
3.0 User Interface	11
3.1 Simulation Area	12
3.1.1 The Globe Display	12
3.1.2 Terrain and Land Cover	14
3.1.3 Unit, Group, and Weapon Symbols	15
3.1.4 Group versus Unit View Mode	18
3.1.5 Upper Game Bar	18
3.1.6 Lower Game Bar	19
3.1.7 Message Log	21
3.1.8 Time Step Buttons	21
3.2 Mouse Functions	22
3.3 Buttons and Windows	24
3.3.1 Engage Target(s) - Auto	24
3.3.2 Engage Target(s) – Manual	24
3.3.3 Plot Course	26
3.3.4 Throttle and Altitude	27
3.3.5 Formation Editor	29
3.3.6 Magazines	30
3.3.7 Air Operations	32
3.3.7.1 Aircraft Status Dialog	32
3.3.7.2 Air Facilities Tab	34
3.3.8 Boat Operations	34

3.3.9 Mounts and Weapons.....	36
3.3.10 Sensors	37
3.3.11 Systems and Damage.....	37
3.3.12 Unit/Group Doctrine	38
4.0 Menus and Dialogs	51
4.1 Right Click on Unit/ Context Dialog	51
4.1.1 Attack Options	51
4.1.2 ASW-specific Actions:	52
4.1.3 Context Menu, Cont.....	53
4.1.4 Group Operations:.....	54
4.1.5 Scenario Editor:	54
4.2 Control and Shift Right Click on Map Dialog:	55
4.3 Units, Groups and Weapons Symbols.....	56
4.4 Group Mode and Unit View Mode.....	56
4.5 Right Side Information Panel	57
4.5.1 Unit Status Dialog.....	57
4.5.2 Sensors Button.....	59
4.5.3 Weapon Buttons	59
4.5.4 Unit Fuel	59
4.5.5 Unit Alt/Speed	60
4.5.6 Unit Fuel	60
4.5.7 Unit EMCON	60
4.5.8 Doctrine	60
4.5.9 Doctrines, Postures, Weapons Release Authority, and Rules of Engagement	60
5.0 ScenEdit.....	62
5.1 Getting Started	62
5.2 Scenario Intro Walkthrough	62
5.2.1 Full Scenario Walkthrough	63
5.2.2 Final Checklist for Polishing.....	63
5.2.3 Detailed Single Scenario Walkthrough	65
5.3 Lua scripting	77
5.4 Editor Drop-Down Menu	81
5.4.1 Scenario Times + Duration.....	81
5.4.2 Database	82
5.4.3 Campaigns	83

5.4.4 Add/Edit Sides	83
5.4.5 Edit Briefing	85
5.4.6 Edit Scoring Dialog.....	85
5.4.7 God's Eye View	86
5.4.8 Weather	86
5.4.9 Scenario Features + Settings.....	87
5.4.10 Merge Scenario	89
5.5 Events	89
5.5.1 Actions	90
5.5.2 Special Actions	92
5.5.3 Triggers	92
5.5.4 Setting up Triggers.....	94
5.5.5 Constructing an Event.....	96
5.6 Scenario Batch-Rebuilder (SBR).....	96
5.7 Unit Actions	96
5.7.1 Add a Unit	96
5.7.2 Add Satellites.....	98
5.7.3 Edit Aircraft.....	100
5.7.3 Add/Removing a boat or submarine:.....	101
5.7.4 Edit Docked Boats	101
5.7.5 Remove All Units of the Side	102
5.7.6 Edit Cargo	102
5.7.7 Other editor functions	103
5.7.8 Publish Scenario to Steam Workshop (Steam only)	103
5.8 Import/Export Units.....	104
6.0 Drop-Down Menus	106
6.1 File	106
6.2 View.....	107
6.3 Game.....	108
6.3.1 Start/Stop	108
6.3.2 Time Compression	108
6.3.3 Order of Battle	108
6.3.4 Database Viewer	110
6.3.5 Browse Scenario Platforms	112
6.3.6 Scenario Description.....	112

6.3.7 Side Briefing	113
6.3.8 Side Doctrine/ROE/WRA/EMCON	113
6.3.9 Special Actions	113
6.3.11 Message Log	114
6.3.12 Losses and Expenditures	114
6.3.13 Scoring	115
6.4 Game Options Window	117
6.4.2 Map Display	117
6.4.3 Message Log	118
6.4.4 Sounds and Music	118
6.4.5 Game Speed	118
6.4.6 TacView	118
6.4.7 Hover Info	119
6.5 Map Settings Drop-Down Menu	119
6.8 Contacts Drop Down Menu	123
6.9 Missions & Reference Points Drop-Down Menu	123
6.9.1 Mission Editor Menus	123
6.9.2 Reference Point Menus	123
6.9.3 No Navigation and Exclusion Zone Menus	126
6.9.4 Quick Creation and Editing of Zones on the Map	127
6.10 Help Menu	128
7.0 Missions, Cargo Missions, Operations Planning, and Reference Points	129
7.1 Mission Editor	129
7.1.1 Adding a new Mission	129
7.2 Mission Definitions and Details	136
7.2.1 Ferry Mission	137
7.2.2 Support Mission	138
7.2.3 Patrol Mission	139
7.2.4 Strike Mission	141
7.2.5 Mining Mission	147
7.2.6 Mine-Clearing Mission	148
7.2.7 Cargo Mission	148
7.2.8 Cargo / Edit Cargo	150
7.2.8.1 Creating a Sequence of Cargo Missions	151
7.2.8.2 Units of Measure in Cargo Editor	152

7.3 Reference Points.....	152
7.3.1 Changing Reference Point Properties.....	153
7.4 Operations Planning	154
7.4.1 Nomenclature to understand	154
7.4.2 Understanding the concept of dynamic units	155
7.4.3 Mobile facility vs Ground unit and Split/Merge ground units	157
7.5 Operation Planner	159
7.5.1 Mission priority	160
7.5.2 Mission phases	161
7.5.3 Column Definitions.....	161
7.5.4 Triggers	162
7.6 Air Tasking Order.....	167
7.7 Benchmark Mode	168
7.8 Lua Reference	169
8.0 Databases and Templates.....	173
8.1 Scenarios vs. Databases	173
8.2 Scenario Maintenance	173
8.3 Rebuild Single Scenario	174
8.4 Rebuild Multiple Scenarios.....	174
8.5 Shallow Rebuild vs. Deep Rebuild	174
8.6 Log files	175
8.7 Scenario Config Files (INI file).....	175
8.8 Scenario Config File Templates and Delta Templates.....	176
8.9 Editing Scenario Config Files.....	177
8.10 Scenario Config File Overview.....	177
9.0 COMBAT.....	180
9.1 Sensors.....	180
9.1.1 Radar.	180
9.1.2 ESM	181
9.1.3 Sonar	181
9.1.4 PCLS.....	182
9.1.5 Other (MAD)	182
9.2 Weapons.....	182
9.2.1 Unguided Weapons	182
9.2.2 Guided Weapons	183

9.2.3 Torpedoes	184
9.2.4 Lasers.....	185
9.2.5 Electromagnetic Pulse (EMP)	185
9.2.6 Warhead Types	186
9.2.7 Decoys.....	186
9.2 Battle.....	187
9.2.1 Air Combat	187
9.2.2 Naval Combat.....	188
9.2.3 Submarine Combat.....	189
9.2.5 Land Combat.....	193
9.2.7 Damage and Repairs	194
9.2.8 My @#%^@#% weapon won't fire!!!	194
9.2.9 DLZ And Why It Matters	199
9.2.10 A Note On Losses.....	200
9.3 Construction and Destruction	201
9.4 Unmanned Aerial Vehicles	204
10.0 Example Scenario Play.....	205
10.1 How to Start a Scenario	205
10.2 Wooden Leg, 1985.....	207
10.2.1 Collecting Intelligence and Devising our Strategy	207
10.2.2 Executing our Mission.....	214
10.2.3 Debrief	223
10.3 Battle of Chumonchin Chan, 1950	224
10.3.1 Collecting Intelligence and Devising our Strategy	224
10.3.2 Executing the Mission.....	228
10.3.3 Conclusions and Discussion	242
10.4 Trapped Under Ice, 1999	242
10.4.1 Collecting Intelligence and Devising our Strategy	242
10.4.2 Executing the Mission.....	244
10.4.3 Conclusions and Discussion	255
10.5 AAW Example – To be completed.....	256
11.0 Appendices	257
11.1 Keyboard Commands:	257
11.2 Custom Overlays 101.....	259
11.3 Database Edit Capability in the Scenario Editor	260

11.4 Common Air Units..... 261

 10.4.1 Early/Mid Cold War 261

 10.4.2 Late Cold War/Modern 261

 10.4.3 Aircraft Organization 262

11.5 Common Sea Units 262

11.6 Common Naval Tactics..... 263

11.7 Comms Disruption & Cyber Attacks..... 264

11.8 Tacview 268

11.9 Detailed Design Notes 268

12.0 Glossary 310

13.0 COMMAND Update History 311

FAQs, Resources, and Learning: 378

 CMO Architecture 378

 Learning CMO 378

CREDITS..... 380

Preface

Congratulations! You have purchased COMMAND: MODERN OPERATIONS, the powerful global simulation of modern warfare at sea, in the air, in space and over land. While the COMMAND development team have done their best to make COMMAND'S user interface and gameplay intuitive, your gameplay experience will be greatly enhanced by reading this manual.

Historical Introduction

After the signing of the Japanese Instrument of Surrender in 1945, the naval world had changed forever. Of all the grand fleets of the past, only the United States and its former colonial master, now inseparable, stood intact. The Americans ruled over the sea as no power had before. The Victorian Royal Navy talked about a "Two-power standard" of matching the two next fleets. The Americans now enjoyed an "every power standard" courtesy of their smashing victory, a standard they have not been keen on letting slip. As fearsome as the Soviet Union was on land, its sea power was marginal immediately following the war.

The age of the traditional fleet battle was over. Even had Stalin's big fleet program came to fruition, the American head start was simply too great. But there was another counter.

The missile, the submarine, and the jet bomber served as a potential asymmetric counter to the carrier task force. As the Cold War intensified, all three of these found their way into the burgeoning Soviet Navy. Although the world fortunately never had to find out if the "strike complex" could overpower the carrier group, it nonetheless emerged as a distinct threat.

This technological arms race produced some interesting systems. The Komar missile boat that sank the INS Eilat was both revolutionary and quickly obsolete. Anachronisms like HMS Vanguard and the reactivated Iowas stood with oddballs like the carrier-launched U-2.

The aircraft carrier meant that the increasing performance of aircraft could be harnessed in naval warfare, showing that all technologies are interrelated. The battleship had been sunk, with jets and anti-ship missiles finishing off what the propeller-driven carrier aircraft had mortally wounded in World War II. And one of the first major procurement battles of the Cold War, the so-called "Revolt of the Admirals" was over which service would take the lead in deploying the biggest game-changer of all: the nuclear bomb.

Meanwhile, the proxy conflict between the superpowers was throwing gasoline on the fires ignited by the collapsing colonial empires. The story of modern war would be written here. A large intervention here, a small one there. The clash of two regional powers with the best gear their patrons could provide, the dance of two minnows armed with low-grade weapons. The defining conflict of missile-age sea war was fought over two islands. The carrier battles of World War II gave way to one near-miss that would have involved only three carriers at most.

And the ever-present fear of nuclear war, of the Russo-American World War III, of the best of the best smashing against each other in an apocalyptic struggle, dominated everyone's mind.

After a lull in the peace of the 1990s and unconventional war of the 2000s, the rise of China and oil-fueled resurgence of Russia brought the issue of naval technology back to the forefront. Now, the missile age has reached a new level, with weapons such as the Naval Strike Missile, Yakhont, and upcoming ones like the LRASM (Long Range Anti-Ship Missile) and DF-21 ballistic missile reaching new limits of stealth, speed, and range.

That was at sea.

In the air, things were considerably more even. While the immediate postwar Soviet navy was a small afterthought, its air force was not. The jet age of aerial warfare started with a bang in Korea and never let up, with the newly independent United States Air Force having its bloody baptism of fire. The surface-to-air missile added a new element to the old struggle in one direction, and the smart bomb likewise in another.

The period from Korea to Vietnam brought a focus on strategic capability as the superpowers relied on bomber aircraft as their primary deterrent before intercontinental ballistic missiles could get off the ground. This led to some flashes in the pan like the B-58 and M-4 Bison. It also led to enduring designs in use to the present day like the famous B-52 Stratofortress and Tu-95 Bear. The paradox of the chaotic 1945-1960 period for air power was that a “cutting-edge” aircraft could soon turn into an obsolete one-trick pony, while a seemingly less-capable one could stick around long after it was reduced to tossing standoff missiles against a peer opponent. The Stratofortress had conventional capability and endurance that has proved itself in every American war since Vietnam, while the Bear has excelled as a long-range patroller. Such paradoxes can be found across the services countries, with military platforms frequently outlasting their intended replacements.

Lessons of the smaller regional wars, not just Vietnam and Korea, were studied and debated extensively. The Falklands and especially Arab-Israeli wars were scrutinized in massive detail. Exercises such as Red Flag brought an ever-increasing amount of realism to training.

Thankfully, the brutal meatgrinder of aircraft and missiles over central Europe did not come to pass. What did emerge was the Gulf War, a battle with American aircrews facing an opponent with the exact technology that they had trained against for over a decade. The resulting one-sided battle cemented American dominance in the skies, at least for a time.

Yet looking at air or naval power in isolation misses the point. Even as a branch, air or naval operations are simply one part of a bigger whole. And as with all combined arms, the whole is greater than the sum of the parts. Coordinated “jointness” is extremely difficult to achieve. Yet when it is achieved, the result is overwhelming dominance.

This is the world of Command: Modern Operations. Reenact the battles of the wars that did happen, pushing packages of Thunderchiefs over North Vietnam or sending F-117s over Baghdad. Explore the possibility of the wars that didn't happen, be they skirmishes over disputed borders or apocalyptic battles in the Fulda Gap. And speculate on the outcomes of wars that *could* happen, in the Spratlys, Baltic States, or wherever the imagination leads.

What is Command?

COMMAND is a 64-bit, pause-able, real-time joint operations wargame that spans from shortly after World War II to the near future (1940s-2020s). In it, you control integrated weapons systems. The most common of these are ships, submarines, aircraft, and the bases they stage from. Yet COMMAND also allows you to lead other less visible and glamorous but no less important pieces of the puzzle. Satellites, unconventional special or guerilla forces, and cyberattacks are all at your disposal. The goal is to integrate all of those parts into one whole.

You fight on a life-sized globe with meticulously detailed terrain both above and below sea level. Everything from hills to thermoclines is placed on the map and taken into account, and it's possible to stage from one continent, strike another, and return. The instructions given to your forces can be as broad or detailed as necessary. COMMAND offers the ability to shift from hands-off broadly defined missions to hands-on personal control and back at any time.

COMMAND includes a wide range of scenarios that span from brushfire counterterrorism to short but sharp duels of fast attack craft to theater-wide engagements where aircraft and ships across an entire ocean must be coordinated and synchronized. Its internal scenario editor allows for an even greater range. Scenarios on both extremes of the conflict scales have been made, from nonviolent coast guard duties on one end to full-bore nuclear exchanges on the other.

COMMAND is aimed to both entertain and educate. It shows the evolution of warfare from the 1940s to the present and illustrates the complexity that such combat has always entailed. It is a tool to show what has changed throughout the decades and what has not, study the lessons of history, speculate the future, and have fun.

1.0 Installation

MATRIX INSTALLATION BOILERPLATE HERE from ERIK

1.1 System Requirements

Minimum system requirements, with recommended requirements in parentheses:

OS: Vista / 7 / 8 / 10 (*since version 1.05, Windows 7 is not supported*)

CPU: Any multi-core CPU; Quad-core CPU and above recommended

RAM: 2 GB (4GB+ recommended)

Video/Graphics: DirectX 9.0c compatible video card with 128 MB RAM (*since version 1.05, DX11 is STRONGLY recommended*)

Sound: Compatible sound card

Hard disk space: 15 GB free minimum, actual used space likely to grow with usage

DVD-ROM: Yes, for boxed version

DirectX version: DirectX 9.0c (Suitable Direct-X version bundled with game) (*since version 1.05, DX11 is STRONGLY recommended*)

Note: Large scenarios with many “moving parts” will run slower on the same computer than smaller ones.

1.2 Support

The quickest way to get responses from members of either the COMMAND development team or the simulation community is to:

- Register as a new user of the forum at matrix games forums (<http://www.matrixgames.com/forums/>).
- Log in, and find the Command support area.
- (Optionally) Do a quick search for previous posts and threads on keywords best reflecting your problem. The “bug” might turn out to be a by-design gameplay element or have been already found and fixed.
- Start a new thread with a meaningful name and (optionally) a quick reference to an unsuccessful search for “prior art”. As a minimum, please specify the version of COMMAND you are running and the scenario (if there are any problems with gameplay), and the mode of play.
- Should a bug be reported, a save game file of the bug in ‘action’ is ***absolutely necessary***. COMMAND is an incredibly complex simulator, and thus a save is needed to be able to reproduce the bug and provide ways to fix it.
- Wait for replies.
- What to do when editing a scenario: If you need assistance editing a scenario there are resources that will help. We hope this manual will be a primary resource, but we do expect to see questions and posts on our support forums which the developers will monitor along with experienced players.

1.3 Notes for Multitaskers and Returning Players

- COMMAND does not automatically pause when you switch to other applications. The message popup will still occur for those messages set to popup as requested in the Game Options menu.
- COMMAND has undergone many changes from version 1.0 in September 2013 to the present. For a detailed list of each and every change, see 13.0, [COMMAND Update History](#) at the end of this manual.
- As of version 1.3, if the computer goes to sleep while COMMAND is running, it will wake up with COMMAND running.
- Different scenarios can have vastly different performance on the same computer. Scenario editors should be encouraged to limit the number of “moving parts”, such as units not set to “blind”, satellites, or units in general.

2.0 Introduction to COMMAND

Entering COMMAND, there are four main game modes.

Game Modes:

Campaign. This game mode allows for the playing of multi-mission campaigns, such as the official DLCs or tutorials. See 5.4.3, Campaigns, for more information.

Quick Battle. This game mode allows for players to throw together a “quick battle” from a list of presets. Thus you can choose between F-16s and F-22s on one side and Su-35s or MiG-29s on the other for an easy to execute dogfight, or a submarine matchup involving a Los Angeles or Virginia vs. Victor III or Akula.

Start New Game. This loads a prebuilt scenario. This can be considered “normal play”. Players can select a side if the scenario allows it, but all ScenEdit features are disabled.

Create New Scenario. This allows for a scenario to be created. All ScenEdit features are enabled.

The gigantic amount of community content for COMMAND and its complexity may be daunting to beginners. Completing the formal tutorials, some new to CMO and some carried over from 1.0, in order is highly recommended. In addition to those, the following “standalone scenarios” are useful for learning the basics of COMMAND.

- **Battle of Chumonchin Chan, 1950.** This is a basic Korean War scenario where the player controls an objectively superior force to their opponent. Useful for learning gun-dominant surface warfare.
- **Wooden Leg, 1985.** This is another basic long-range Israeli airstrike scenario where the only challenge is aerial refueling to get the aircraft to the target. Useful for learning how to use tanker aircraft.
- **Raid on Kismayo, 2013.** This is a basic low-intensity modern warfare scenario. Useful for general play and control in a generally low-risk environment.
- **Battle of Latakia, 1973.** This is a historically lopsided encounter that serves as an excellent introduction to missile-age surface warfare. Useful for learning missile-dominant surface warfare.
- **Stand Up, 2011.** This is, especially if played from the British side, an easy scenario that nonetheless forces the player to coordinate aircraft and ships. Useful for getting the hang of combined arms operations.

Once the basics have been mastered, the following are recommended for somewhat more advanced play:

- **North Pacific Shootout, 1989.** A small-scale air-to-air scenario between the best the US and Soviet Union have to offer. Useful for fighting a peer air opponent in a manageable situation.
- **Iron Hand, 2014.** A Russia-vs-Azerbaijan scenario that teaches the basics of standoff weapons, a detailed air campaign, and operating in a casualty-conscious environment.

The community pack, available at https://command.matrixgames.com/?page_id=1876, contains hundreds of scenarios available for download. Many are beginner-friendly. A helpful rule of thumb is to search for ones with low ‘complexity’ on the difficulty-complexity scale.

Disclaimer: The community pack scenario authors have varying degrees of contactability and support of their scenarios. Thus, some scenarios may not be updated or changed to take mechanics changes since their introduction into account. Play at your own risk.

2.1 Important Terms

There are several bits of information those in a hurry to get started need to know. These will be elaborated in greater detail throughout this manual and the tutorials.

Units (Page 60): Units are the entities in the simulation. There are many types of units and many of these units can house other units; for example, an aircraft carrier can house aircraft, or an airbase can have hanger units, runway units, and these hanger units may contain aircraft units. Many units can be directly controlled or assigned to missions. There are special units such as individual facilities, sonobuoys, missiles, and many weapon types that may or may not be directly controlled and cannot be assigned a mission. Airbases can be assigned missions.

Groups (Page 62): Groups are units that have been bound/grouped together in the simulation into one cohesive unit. This is done by drag-selecting units and pressing the G key. Many scenarios come with groups pre-made by the scenario creator, and aerial missions automatically group aircraft based on the mission editor's flight size.

Mounts (Page 36): Many units have mounts. These are usually weapons or sensors. An example would be a gun mount on a ship, a hard point on an aircraft, or a sonobuoy dispenser on an anti-submarine warfare (ASW) helicopter. Mounts usually require ammunition that is already loaded, such as a missile on an aircraft (single shot mount) or a gun mount with a magazine associated with it for reload. There are ways to reload mounts and magazines if the scenario designer desires to provide loaded ammunition bunkers, ships' magazines for aircraft or at-sea replenishment for ships.

Magazines (Page 31): These are components of units that are used to store weapons. These must have weapons in them for weapons to fire or aircraft to launch combat-oriented missions.

Missions (Page 130): Missions make the battle space manageable and allows the scenario designer to set up an effective AI controlled opponent. The purpose of missions is to allow units or groups of units to engage in a common task, such as anti-submarine warfare or all striking one target. There are many mission types and mission options, such as Rules of Engagement, Doctrine and Postures.

Formations (Page 29): Formations are a group of units (normally associated with ships, but may be extended to aircraft formations), a central unit(s) and escorts. Each unit in the formation usually has a primary task and station, designed to mutually support the whole formation from any threat.

Rules of engagement (RoE) and Doctrine (Page 38): This also encompasses mission options and orders. Many units house other units, like a cruiser with an embarked ASW helicopter. The cruiser is assigned a mission or patrol zone within a formation which has a specific EMCON (Emissions Condition, ie, whether its radar is on or off). If the embarked helicopter is not assigned a specific mission, it will inherit the Rules of Engagement, Doctrine and EMCON of the parent unit (the cruiser), this can be overridden if you decide to change any of the aforementioned settings (you will be prompted to affirm the override). You can skip right to the tutorials or read further for more detailed information on the items above.

Events (Page 90): Events allow fundamental changes in the battlespace, such as changing the player's score, launching or cancelling a mission, or even ending the scenario entirely. Events consist of a trigger (cue to start them, which can be a point threshold, unit hit/found, or time point) and an action (the meat of the event itself,

which can be everything from a basic “Side ____ scores X number of points” to a complex Lua script). Events are created by the scenario author in the scenario editor and are not viewable or editable in regular play.

Sensors (page 179): Sensors are anything that allows a unit to detect something else. They range from the classic “Mark 1 Eyeball” to advanced radars and electronic emission detectors. Sensors can be passive (no emissions of their own) or active (which do emit, and thus can be detected by other passive sensors)

Types of sensors include:

- **Radar.** The most famous ‘active’ sensor system. Radars range from the earliest and crudest World War II vintage types to ultra-modern AESAs (Active Electronically Scanned Arrays). Radar performance, especially in the face of jammers, can thus vary extensively based on power and technology. On one extreme are early radars that can only generate an imprecise general contact location, while on the other are advanced ones that are incredibly jamming resistant and can determine the exact type of target they’re identifying.
- **Sonar.** Sound-based naval detection system, sonar can be ‘passive’ (purely listening) or active (pinging). Active sonar is more precise but lets the target know that someone else is there. As with radars, sonar performance varies extensively by size and technology.
- **Electro-Optical.** This includes video cameras and infrared sensors. While they have their limitations (such as only being able to determine precise distances at short range), they have the advantage of being ‘passive’ and thus not emitting like radars are.
- **ESM.** Electronic Support Measures, these range from simple radar detectors to advanced signals processors. All share the characteristics of being ‘passive’ (not emitting themselves) and being able to detect active radar emissions.

2.2 Fundamentals

2.2.1 Starting COMMAND

Start command by either clicking on the Desktop Shortcut or on the Command.EXE file in the Command folder. The screen below is shown.



On the top is the “play game” subsection.

“**QUICK BATTLE**” opens a series of customizable engagements where the player selects the forces from a group of presets and proceeds to engage them in a small scenario.

To create a quick battle involves templates. They are accessible in the [COMMAND installation folder]/QuickBattle folder. Each quick battle gets its own subfolder, featuring an HTML document and a Lua script to match.

In the background, the named elements in the HTML page are translated to Lua variables which are then executed via the attached script. Thus, like scenarios, the templates can be customized and shared by editing the template portions.

“**CAMPAIGNS**” opens a list of current campaigns. (See 5.4.3, Campaigns)

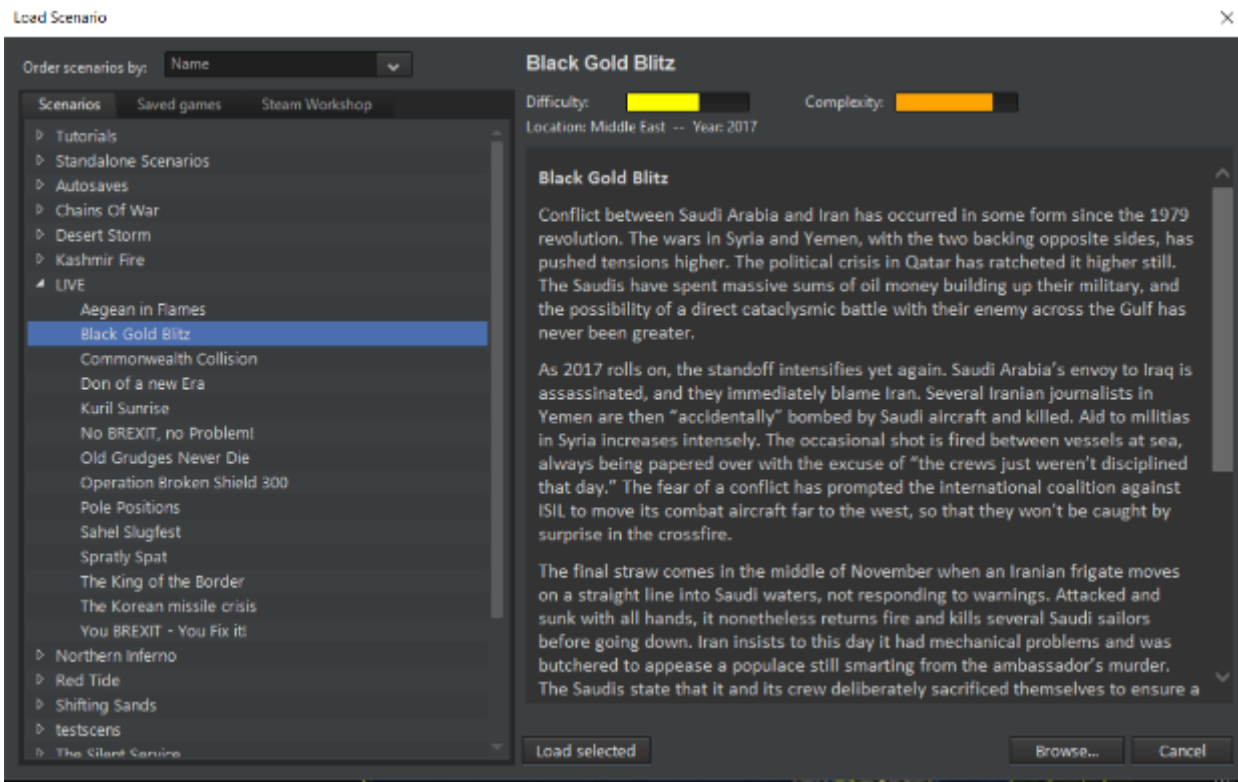
“**START NEW GAME**” and “**LOAD SAVED GAME**” places the simulation in play mode and opens a menu with two tabs on the left and a file menu below them. The “**START NEW SCENARIO**” tab shows scenario files and allows starting the selected scenario from the beginning and the “**LOAD SAVED GAME**” tab shows saved-game files and allows loading those saved games. Selecting a scenario or saved game will show the title and a brief description of the scenario.

RESUME FROM AUTOSAVE loads the last autosave.

On the bottom is the “Scenario editor” subsection.

The “**CREATE NEW SCENARIO**” places the simulation in the edit mode and starts the simulation in a blank mode.

The “**EDIT SCENARIO**” places the simulation in the edit mode and allows loading a scenario or saved game.

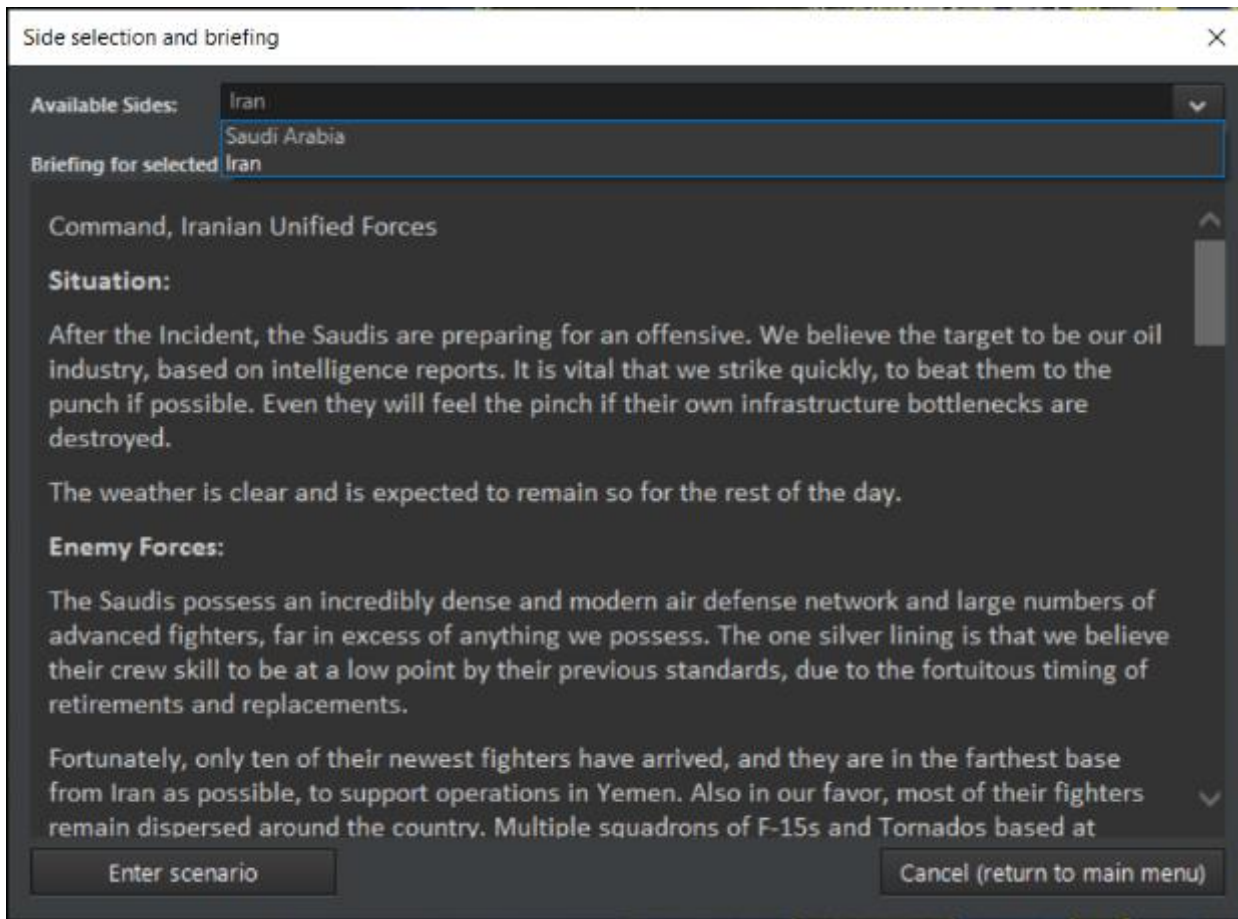


The load scenario dialog is composed of list of scenarios or saved games on the left (with a third section on the rightmost side of that reserved for scenarios accessed via the Steam Workshop) and a large text box with name and description on the right. The left scenario drill-down list is tabbed and allows you to view and choose scenario folders, files (scenarioname.scen) or save games (scenarioname.sav files).

To load a scenario or save game you must select one by point and clicking it from the left side menu. If the entry has a plus sign next to it is a folder which when clicked will open revealing individual scenario or save game files that can be individually selected. When satisfied, click the load selected button at the bottom and the scenario or save will load.

When you load a scenario in either mode the application will match the scenario with the necessary database version. When you edit a scenario, it will start with your default database version. If you need to change this, you can do so in the scenario editor. This will be described later in the manual.

You will now be presented with the side selection and briefing dialog.



Select a side in the drop-down menu at the top, read the briefing and click the “enter scenario” button when satisfied with your choice. Many scenarios do not have this option. In this case, simply read the briefing if desired and press “enter scenario”.

The scenario will now launch, along with the features and settings graphic that determines what specific options are enabled or disabled. Some scenarios may have unlimited magazines at air and naval bases, while others may not.

In normal play, scenario features and settings are locked. See 5.4.9, “Scenario Features + Settings” for information on how to change them in the editor.

3.0 User Interface

Command's user interface (UI), in simplest terms, is a zoomable globe bordered by a menu, game controls and informational displays. The globe allows for action on a truly worldwide scale. Units can move from one hemisphere to another (think intercontinental missile launches or B-2 raids from Whiteman AFB in the central United States to Serbia or Iraq and back), following a proper curved path as they do so. Actions such as REFORGER convoys or the reinforcement of East Asian bases can be realistically simulated in this large-scope system.

3.1 Simulation Area

3.1.1 The Globe Display



Viewing and manipulating the scenario map is simple. Using a mouse, you can point and right-click to rotate and use the mouse wheel to zoom in and out. You can use the “View” menu item at the top of the application to change the information displayed on the globe, including turning on and off the various range rings and enabling/disabling the many layers included with the game, making the map anything from the default ultra-detailed Sentinel-2 satellite imagery to a blank wireframe. You can also add your own imagery as an overlay by using a third-party program to grab the imagery and pressing the custom overlay button to display it. Personal view preferences can be saved in the Game Options menu. The globe is rendered using DirectX 11 for performance purposes.



The diversity of map options in COMMAND is shown in these two snapshots of the Congolese/Cabindan coast. The top screenshot has the Sentinel-2 layer enabled, while the lower one has the BMNG Layer disabled entirely.

As in the real world, COMMAND's environment is affected by the terrain and weather and has a profound effect on your sensors, weapons, and tactics. It thus pays a lot to be aware of your natural surroundings. To get a better understanding of the world in your operations theater, feel free to mouse over the globe and notice the cursor data block's display of land elevations and terrain types while over land and bathymetric data (including convergence zone [CZ] information) while over water.

This is important because terrain affects operations greatly. Underwater, the data block displays the depth of the water and other features vital for submarine warfare. One is the depth and strength of the thermal layer, a

feature that serves as a natural barrier to sound transmission. Another is the distance of convergence zones, which allow vessels with long-range passive sonar to pick up contacts well beyond the short range of direct paths.

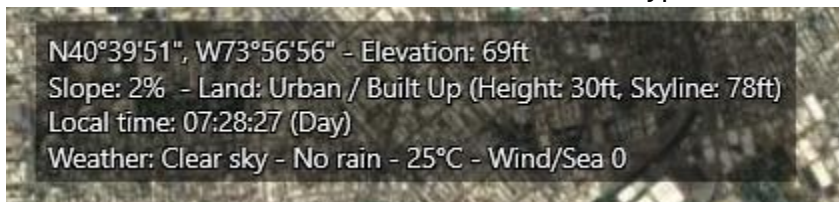
Over land, the slope and type of land determines how fast land units can move, what exact paths they take (units will steer around mountains rather than trying to climb them), how visible they are, and how shielded they are from explosive attacks against them. Greater slope, even without any terrain effects, makes land units slower. Water terrain features (lakes) are totally impassable by land units.

Wetlands and snow both slow down land units considerably (the former to a greater extent) and make them less visible. Forests, and to a lesser extent shrublands, both significantly slow down and shield the land units from eyes and bombs alike. Croplands and vegetation mosaics have a small visibility and speed reduction, while savannas have a minor speed reduction. Barren and grassland terrain acts as a “base case” with no modifiers, while urban terrain offers the greatest shielding and, contrary to the others, makes land units slightly *faster* due to the availability of paved roads. Several terrain types are also highly disruptive to radar detection. These types are: forests (all sub-types), croplands and urban. Obtaining a radar detection in these areas is significantly more difficult.

Moving the cursor around the theater can be very important. The presence or absence of convergence zones is a game changer in anti-submarine warfare. As sloping and terrain significantly determine the pathing of land units (and the presence of islands and shores the pathing of naval ones), seeing what’s where is important in knowing where to send one’s bomber aircraft to hit those land units.

3.1.2 Terrain and Land Cover

COMMAND simulates many different effects of terrain ranging from unit detectability to the effects of munitions on in dense and barren areas. In addition to the type of terrain, terrain has three height values:



Elevation refers to the ground level of the terrain with no objects present such as trees or buildings. The height refers to the average height of objects on that terrain, such as trees or buildings, and the skyline is the average height of the objects plus the base level of the terrain underneath it. These values are useful to a planner because they give you the ability to plan terrain masking attacks, and some types of units such as AH-64E Apache Guardian helicopter can ‘rest’ behind the skyline of the terrain and stick various sensors outside of the terrain making it possible to detect units without being easily engaged.

The slope of terrain as well as the type of terrain has an impact on the performance of ground units. Some units such as hovercraft can easily move between different types of terrain, but struggle with slope and certain terrain densities. When planning ground combat, it is important to consider this along with visibility in order to best deploy your units.

3.1.3 Unit, Group, and Weapon Symbols

Units can be utilized in the game as individual units or groups. Groups are useful to create as you can give many units common orders and you can organize ships into formations for a better common defense.

To group units simply drag-select the units you would like to group together and press the G key. You will now notice they have merged into one group symbol, which you can rename to whatever you would like.

If you'd like to remove members or dissolve the group:

- In game mode, you can detach individuals by select then in unit view (see below) and pressing the d key. Once you get down to one unit the group will dissolve.
- In editor mode you can select the group and press the delete key once which will dissolve the group.

Units and Groups are displayed on the game map as color-coded symbols. The colors note postures (player side, friendly, unfriendly and hostile) and the symbols themselves let you know what kind of units they are:

	Player Side and Friendly*	Neutral	Hostile	Unfriendly	Unknown
Aerostat					
Air Decoy					
Fixed Wing Aircraft					
Helicopter					
Air Group					
Air Base					
Installation					
Mobile Unit					
Mobile Group					
AAA					
SAM					
Radar Site					
Armored Unit					
Infantry					

Mechanized Infantry					
Gun Artillery					
Rocket / Missile Artillery					
Surface Unit: Aircraft Carrier					
Surface Unit					
Submarine					
Surface Group					
Subsurface Group					
Missile					
Torpedo					
Satellite					
Active Sonobouy					
Passive Sonobouy					
Unknown					

* Allied sides are marked with an "A" under them; one-way friendly sides (e.g. you are friendly towards them but they are neutral towards you) are marked with an "F" under them.

Unguided Bomb	
Depth Charge/RBU	
Unguided Rocket	
Bottom Mine	
Floating Mine	
Moored Mine	
Mobile Mine	

Unit Aimpoint Counts: The small white numbers next to units are a mount and sensor aimpoints within the unit. This allows players to know how many components (ex. Each tank of a 4 point tank platoon) are within a unit so they can allocate the correct number of weapons).

Reference Points: Reference points are markings that can be used as area boundaries, waypoints or simple markers. There are several types with different properties:

- Unselected/Unlocked Reference
- Selected Reference Point
- Locked Reference Point

3.1.4 Group versus Unit View Mode

You can toggle between unit view and group view by pressing the 9 key on the keypad of your keyboard. To check which mode you are in please check the value on the bottom left hand side of your display.

Group view is default and displays the group symbol and group datablock for units that have been grouped together. You can see individual units in a group when it is selected but cannot select and give individual orders to ships.

Unit view displays every unit and datablock of every unit on the map. You can select and give individual units orders in this window.

3.1.5 Upper Game Bar



Two layers of game controls are located below the menu items on the map. The upper bar currently includes the following:

- The **Time Compression drop-down menu** lets you set the speed at which the simulation moves forward. The options are 1x (real time), 2x time acceleration (twice as fast), 5x, 15x, and the flame/turbo option, which moves as fast as your PC can handle the scenario. There is one option beyond flame called double flame or coarse fidelity. When this is selected, the scenario will change the time-step to only make calculations when required and thereby move at a much quicker rate. This feature will automatically disengage when high accuracy is required such as during weapons engagements. The actual speed of turbo/flame is defined by the size/complexity of the scenario and the configuration of your PC. The same scenario can run faster or slower on different computers, and the same PC can run different scenarios at different speeds. (Add example?)
- The **Start/Stop button** pauses or resumes the simulation.
- The **Record** button activates Commands internal recording ability. You can use this to view or play files later. Once pressed it becomes the stop recording button. The files this generates are only capable of being viewed in Command and can be quite large.
- The **Custom Overlay** button lets you add imagery from 3rd party sources to the display. See section 10.2 of this manual for specific instructions on how to do this.
- The **Show/Hide Shortcuts** button toggles the lower shortcut bar on and off.
- The **Special Actions** button brings up, if any, the available special actions for the current side.
- The **Options** button brings up the game options menu (see Section 6.4)

You will notice game information displayed at the top border of the globe display. This information includes game time in GMT, local time, remaining time in scenario, the camera zoom altitude in meters, AU count (unit count) and a pulse time (game speed indicator).

Note that the “AU count” and “Pulse Time” numbers are diagnostic settings that are not turned on by default. You can turn them on in game options by checking the Game Diagnostics Check box. Pulse time can be effectively used for bug reports or seeing if one action slows the game down more than another. AU count is useful to determine just how big the scenario can be.

3.1.6 Lower Game Bar

The lower bar currently contains shortcuts for the following actions. Attempting to engage them while having selected an incompatible unit (a unit not under your control in a general sense, or the “Aircraft Operations” button for a unit that holds no aircraft) will simply result in nothing happening. Hovering the mouse over any location on the shortcut bar will display a hotkey that can be used for the same function. These actions are the most critical in COMMAND and are covered in detail in the “Buttons and Windows” and “Mission Editor” sections.



- Engage Targets (Auto): See 3.3.1, Engage Targets (Auto)
- Engage Targets (Manual): See 3.3.2, Engage Targets (Manual)
- Plot Course: See 3.3.3, Plot Course
- Throttle/Altitude: See 3.3.4, Throttle/Altitude
- Formation Editor: See 3.3.5, Formation Editor
- Magazines: See 3.3.6, Magazines
- Aircraft Operations: See 3.3.7, Air Operations
- Boat Operations: See 3.3.8, Boat Operations
- Mounts and Weapons: See 3.3.9, Mounts and Weapons
- Sensors: See 3.3.10, Sensors,
- Systems and Damage: See 3.3.11, Systems and Damage
- Unit/Group Doctrine: See 3.3.12, Doctrine
- Mission Editor: See 7.1 Mission Editor

Note: Despite an attempt to fit the variety of computers running COMMAND, windows may appear distorted or cut short due to DPI issues. If that appears, close the program and run the 4KFix tool located in the Command folder.

3.1.7 Message Log



The message log is in the lower left-hand corner of the screen. It is divided into several subsections containing different types of messages that can be minimized or emphasized. This is so that a player wanting to see, for example, when their aircraft are taking off or landing, is not overwhelmed by one endgame calculation from a lengthy naval bombardment after another.

Moving the mouse over a message highlights the subject of that message on the map by creating a balloon near it with the message in question. This can be helpful for say, pinpointing the location of a new contact.

Clicking on the plus sign to the left of a message elaborates details on that message. For example, “Bogey #20 now type-classified: UCAV” turns into “Contact: BOGEY #20 has been type-classified as: UCAV (Classification by: Fishbed #1 [Sensor: Mk1 Eyeball] at 3.6 nm)”

The message log can be placed in a separate window by pressing CTRL-SHIFT-M or via a menu (see 6.2, View, or 6.4, Game Options Window). In a separate window the message log can be toggled between the “interactive” view (which has the different message categories and hover-balloons) and the “raw text” view found in previous versions of COMMAND (which simply has a list of detailed text messages arranged in chronological order).

3.1.8 Time Step Buttons

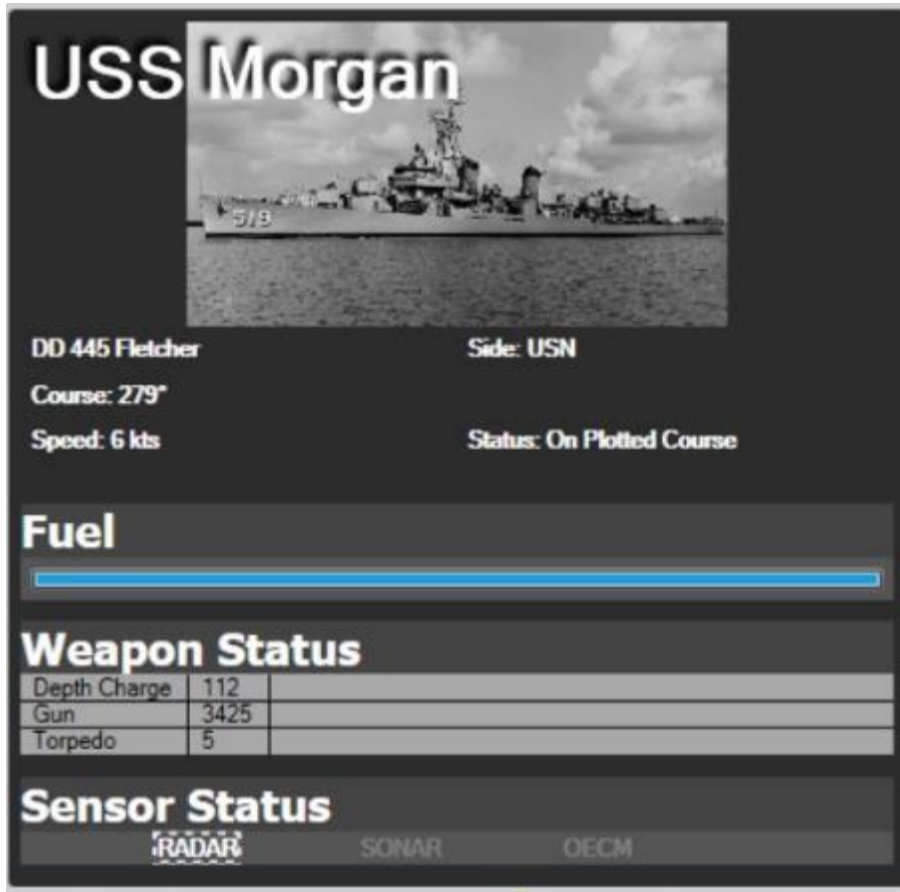
The time step buttons are below the message log and serve as a way of precisely fast-forwarding COMMAND in intervals that range from 15 in-game seconds to 15 minutes. **When the simulation is running, pressing them does nothing.** When the simulation is paused, pressing one of them runs the simulation at the current level of time acceleration for the marked duration and then pauses it again. To move quickly, setting time acceleration to “Turbo/max speed” (the flame icon on the time acceleration menu) before pressing a time step button is recommended.

3.2 Mouse Functions

The mouse has several further functions in the UI.

First you will notice the cursor has a data block next to it which displays information on the environment of the current area it is pointed at. Depending on whether it is over land or sea this information includes position in latitude and longitude; elevation and depth in meters; convergence zone (CZ) layer information (sonar model), terrain slope and type, local time and weather.

If the CTRL key is held down when hovering the cursor over a unit or contact, an additional box appears with more information about the subject in question.



The screenshot shows a detailed information panel for the USS Morgan. At the top, the ship's name "USS Morgan" is displayed in large white text. Below the name is a grayscale image of the ship at sea. Underneath the image, several key statistics are listed in two columns: "DD 445 Fletcher" and "Side: USN" on the first line, "Course: 279°" and "Status: On Plotted Course" on the second line, and "Speed: 6 kts" on the third line. Below these statistics is a "Fuel" section with a blue progress bar that is nearly full. This is followed by a "Weapon Status" section containing a table with three rows: "Depth Charge" with a value of 112, "Gun" with 3425, and "Torpedo" with 5. At the bottom is a "Sensor Status" section with three toggle buttons: "RADAR" (which is currently active and highlighted with a dashed border), "SONAR", and "OECM".

USS Morgan

DD 445 Fletcher Side: USN

Course: 279° Status: On Plotted Course

Speed: 6 kts

Fuel

Weapon Status

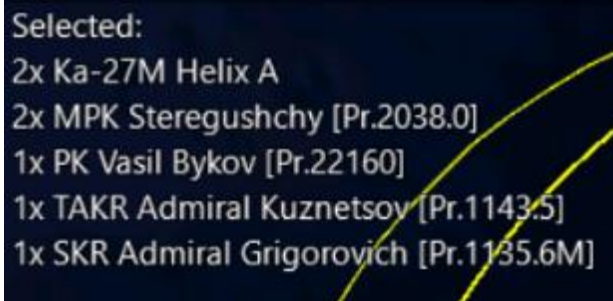
Depth Charge	112
Gun	3425
Torpedo	5

Sensor Status

☒ RADAR ☐ SONAR ☐ OECM

The Fletcher-class destroyer USS Morgan has full fuel, 112 depth charges, 3425 rounds of gun ammunition, and 5 torpedoes in it, and has its radar turned on.

Selecting several units will cause a list, with the units sorted by class and type, to appear in the upper left portion of the map.

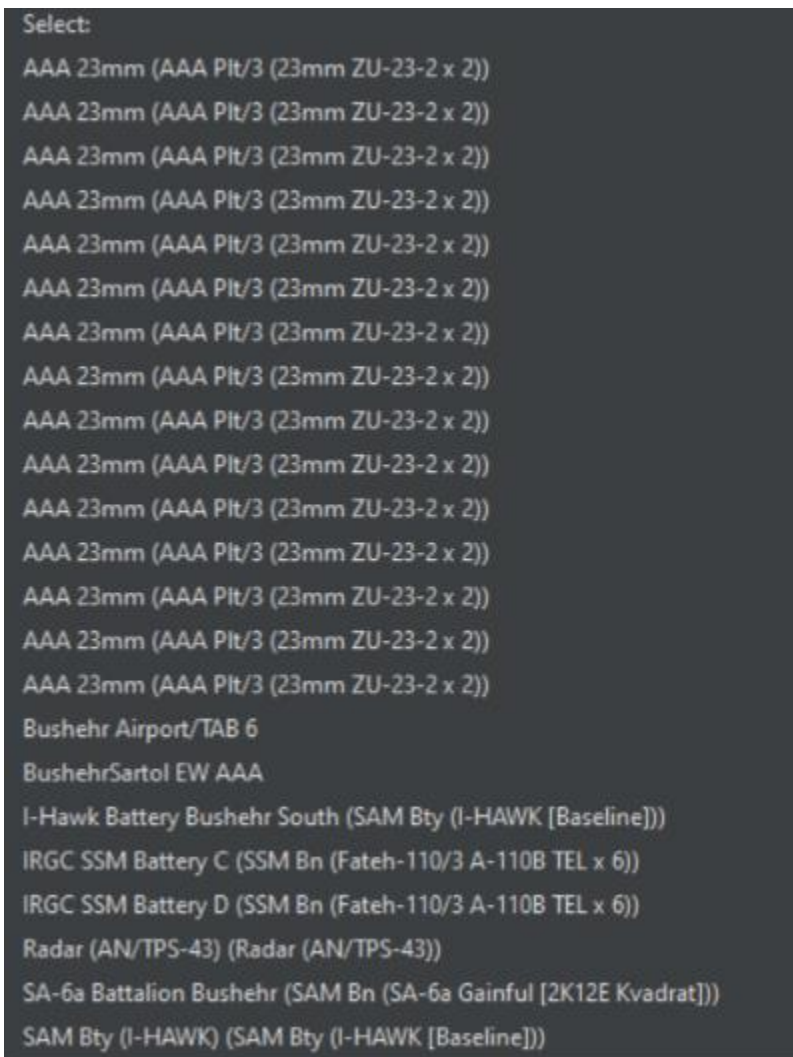


Selected:

- 2x Ka-27M Helix A
- 2x MPK Steregushchy [Pr.2038.0]
- 1x PK Vasil Bykov [Pr.22160]
- 1x TAKR Admiral Kuznetsov [Pr.1143.5]
- 1x SKR Admiral Grigorovich [Pr.1135.6M]

In this example, a Russian carrier group has been selected. Two airborne Ka-27 helicopters appear on top, followed by two Steregushchy frigates and one each of the other three units, including the carrier itself.

When clicking on a unit in close proximity to others, a menu of all the units in the area appears, enabling the player to select any one in particular without needing to zoom in to a large degree.



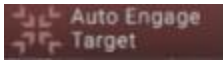
Select:

- AAA 23mm (AAA Plt/3 (23mm ZU-23-2 x 2))
- AAA 23mm (AAA Plt/3 (23mm ZU-23-2 x 2))
- AAA 23mm (AAA Plt/3 (23mm ZU-23-2 x 2))
- AAA 23mm (AAA Plt/3 (23mm ZU-23-2 x 2))
- AAA 23mm (AAA Plt/3 (23mm ZU-23-2 x 2))
- AAA 23mm (AAA Plt/3 (23mm ZU-23-2 x 2))
- AAA 23mm (AAA Plt/3 (23mm ZU-23-2 x 2))
- AAA 23mm (AAA Plt/3 (23mm ZU-23-2 x 2))
- AAA 23mm (AAA Plt/3 (23mm ZU-23-2 x 2))
- AAA 23mm (AAA Plt/3 (23mm ZU-23-2 x 2))
- AAA 23mm (AAA Plt/3 (23mm ZU-23-2 x 2))
- AAA 23mm (AAA Plt/3 (23mm ZU-23-2 x 2))
- AAA 23mm (AAA Plt/3 (23mm ZU-23-2 x 2))
- AAA 23mm (AAA Plt/3 (23mm ZU-23-2 x 2))
- Bushehr Airport/TAB 6
- BushehrSartol EW AAA
- I-Hawk Battery Bushehr South (SAM Bty (I-HAWK [Baseline]))
- IRGC SSM Battery C (SSM Bn (Fateh-110/3 A-110B TEL x 6))
- IRGC SSM Battery D (SSM Bn (Fateh-110/3 A-110B TEL x 6))
- Radar (AN/TPS-43) (Radar (AN/TPS-43))
- SA-6a Battalion Bushehr (SAM Bn (SA-6a Gainful [2K12E Kvadrat]))
- SAM Bty (I-HAWK) (SAM Bty (I-HAWK [Baseline]))

An extreme example of bunching. With two clicks, the player can select the airbase at Bushehr, a search radar by the airbase, one of the four ballistic missile batteries, one of the two SAM units, or one of the many AAA facilities there.

3.3 Buttons and Windows

The functions accessible on the top “shortcut bar” contain most of what one needs to run COMMAND.



3.3.1 Engage Target(s) - Auto

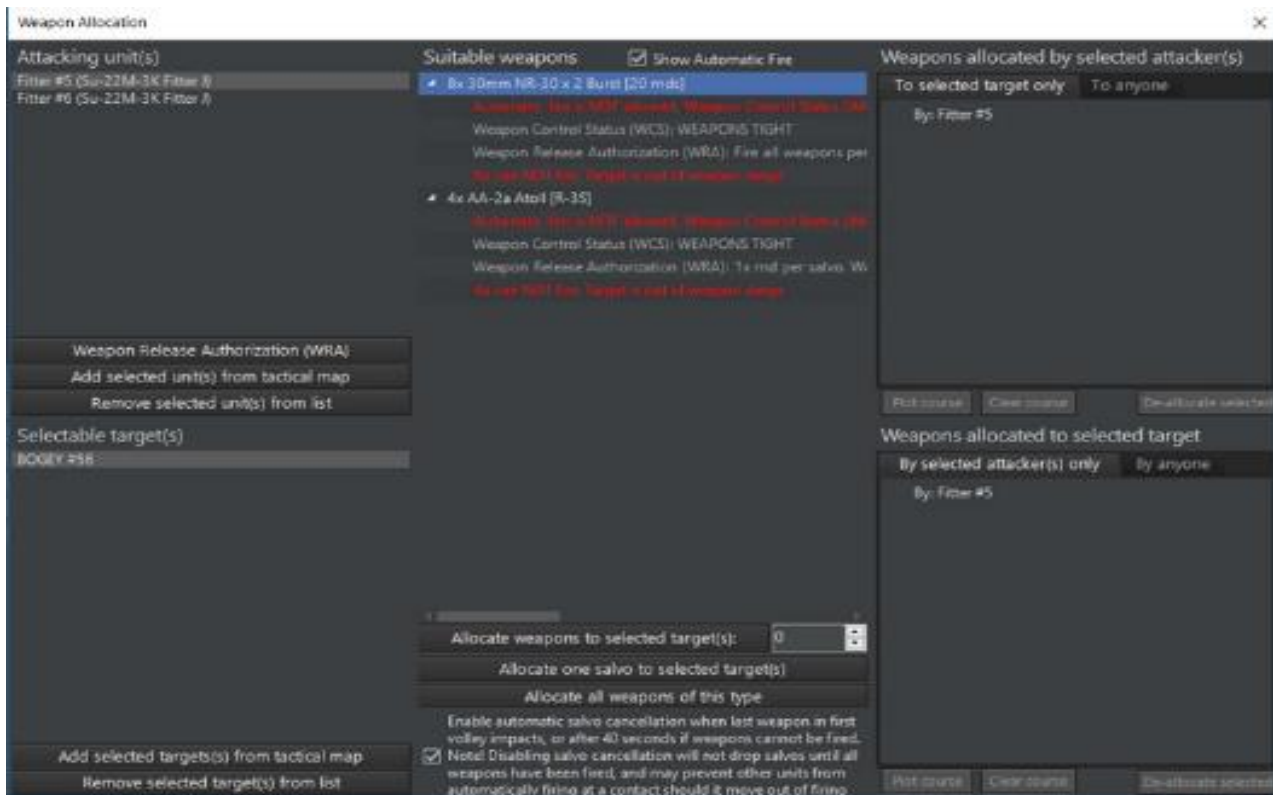
This function will order the selected unit or group to attack a designated target or group of targets. After selecting this function, designate a target by clicking on a target or drag selecting one or more. The mouse cursor will turn into a crosshair to symbolize that the “engage targets” mode has been engaged. The firing unit or group will then move to get within the firing parameters of the weapon to be used. If there is more than one weapon the unit will work from its longest ranged weapons to its shortest (i.e., a warship will fire its missiles at the target and then, if it survives, close to gun range). You can also call on this function using the F1 hotkey or from the right click/unit orders menu. Note that when multiple attackers are given the task of attacking multiple units, their attacks are queued in parallel, that is nits now perform the salvo evaluations in their own individual threads and submit "firing proposals" to their side. The parent side groups firing proposals per target and selects the most promising one, based on criteria that depend on the nature of the target (e.g. for aerospace targets an important factor is time-to-impact).



3.3.2 Engage Target(s) – Manual

This function calls on the weapons allocation dialog, which allows players to manually allocate weapons to selected targets. This is a great function to use in conjunction with plotted courses to allow for off-axis attacks or more fine-grained strike planning; however, you must manually put your units in the correct firing parameters for weapons to fire.

To use, select the function (the cursor will turn into a crosshairs) and then click or drag select one or more targets. The weapons allocation dialog will then launch allowing you to allocation weapons to targets. This function can also be called using the Shift+F1 key combo or from the right click/unit orders menu.



The dialog is organized by having attackers, their weapons and allocations listed at the top and the selected target list and allocations to them listed below.

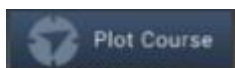
To allocate a weapon to a target:

1. First select one target from the target list below by clicking on it so it is highlighted.
2. Select an attacker from the list above. If they have a legal weapon to fire on the target it will appear in the suitable weapons list group by weapon type. If the weapon is legal but outside on or more firing parameters, the reason will be written in red text (see complete list of reasons and solutions below). If the weapon is within firing parameters, you will see green text saying "OK - Can fire."
3. You then allocate the number of weapons to the target by populating the allocated field below the suitable weapons list and clicking the allocated button or you can double click any weapons line in the suitable weapons list. If you are not sure how many weapons to allocate, please review both weapons and targets in damage values as well as the defender in the platform display. Once allocated, the weapons will appear in both the **Weapons Allocated By Selected Attacker(s)** list at the top and the Weapons Allocated To Selected Target at the bottom. Some weapons are "palletized", and can thus be assigned either individually (ie, missiles at different targets from the same pallet) or all at once (every missile on the pallet is destined for the same target) When launching palletized weapons, it is important to launch every from the pallet in order to achieve maximum effect.
4. Certain missiles can make off-axis attacks. Go to '**weapons allocated by selected attacker(s)**', press the '**plot course**' button, and then '**draw**' an appropriate path. This is highly useful to avoid giving away

one's position and taking advantage of terrain. The courses can be reset by clicking the '**clear course**' button.

5. If you'd like to de-allocate weapons you can do so by double-clicking the individual weapon's lines in the **Weapon's Allocated by Selected Attacker(s)** list.
6. When are satisfied, you can complete the process by clicking on the red "x". You would then have to ensure the firing platforms remain in firing parameters by manually plotting, adjusting speeds and assigning them to missions.

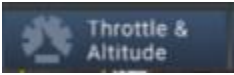
Note: If you are told under the weapon in the weapon list that you aren't allowed to use that weapon, see section 9.2.8 for guidance on how to figure out weapon firing parameters.



3.3.3 Plot Course

A unit's course is a single waypoint at the destination or a series of waypoints to a destination. The unit's current course is displayed on the map as line segments with waypoints designated by small boxes at the nodes.

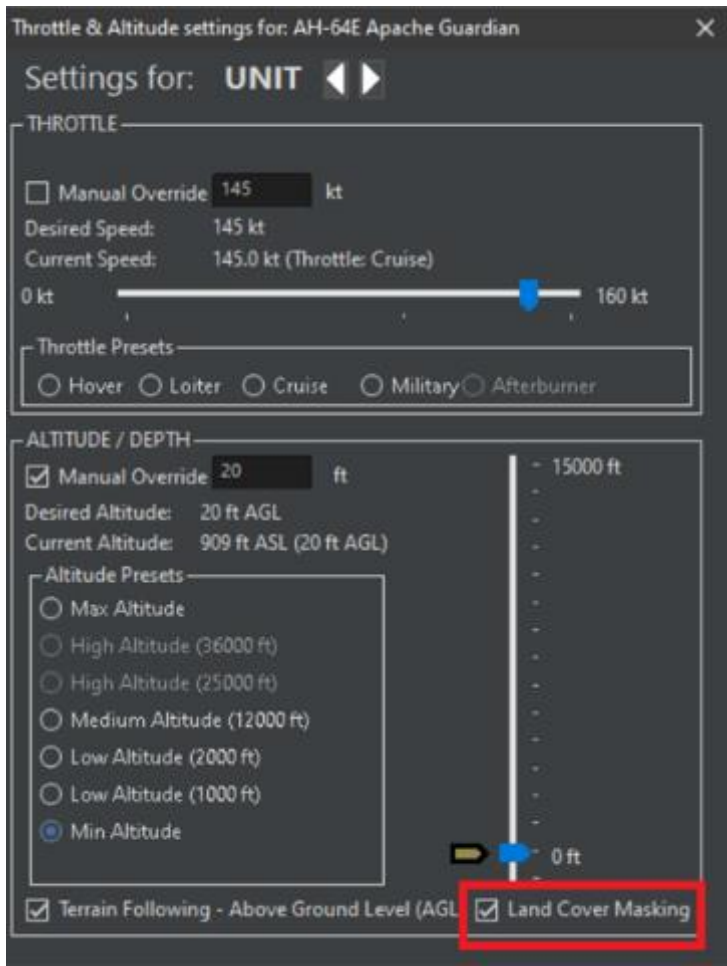
- **Set Course:** Select a unit or group and either press "F3" or select Plot Course on the Unit Orders drop down menu. Click on the map at the desired waypoint location. Continue to add waypoints as desired until the final waypoint has been plotted, then click "F3" or Plot Course to finish.
- **Edit Course:** To edit or change a course or move a waypoint, "Left Click" on a waypoint and drag it to the desired location which will create a new waypoint.
- **Adding Additional Waypoints:** Select a unit or group. Does not work with multiple units/groups selected. Left click on the last waypoint or any waypoint and press F3. Left click on the map to place additional waypoints. Press "Esc" after the last waypoint is placed.



3.3.4 Throttle and Altitude

Throttle/Altitude: Select a unit(s), group, or previously plotted waypoint and either press “F2” or select Throttle / Altitude on the Unit Orders drop down menu. The speed and altitude/depth menu opens. Maximum and Minimum values of speed or altitude/depth are set by the unit’s capabilities and terrain.

- **Unit/Waypoint Settings:** On top of the throttle/altitude window is an arrow set that enables scrolling between unit and waypoint settings. When different settings are applied to the unit, it will change speed/altitude immediately. When applied to a waypoint, it will only do so once the unit reaches the waypoint in question.
- **Speed:** Either select one of the preset throttle settings or use the slider to select the desired speed. The unit(s) will respond to the command immediately. Select one of the presets or use the slider to set the desired altitude/depth. If the unit’s altitude/depth is set by an assigned mission it may be overridden by checking the Manual override checkbox.
- **Altitude/Depth:** Either select one of the presets or use the slider to set the desired altitude/depth. If the unit’s speed is set by an assigned mission it may be overridden by checking the Manual override checkbox. The height of clouds will appear on the slider. Checking “Terrain Following” means that the (air) unit will fly at the specified height above *ground* level rather than above sea level (which can be crucial if the scenario is in a mountainous area and is needed for units with terrain-following capability like the F-111 to use their full potential). Checking “Land Cover Masking” will make the unit attempt to remain at minimum altitude and use the terrain to cover their movement. Think of terms of flying between buildings, under power lines and between trees. Not every flying unit is capable of doing this at such a low level, and its effectiveness will be dependent on the type of unit.



Note: Being as high as possible for aircraft, deep as possible for subs, and fast as possible for ships is not necessarily the best way to handle units. Pay attention to mission, sensors, and threats before setting throttle and altitude/depth.

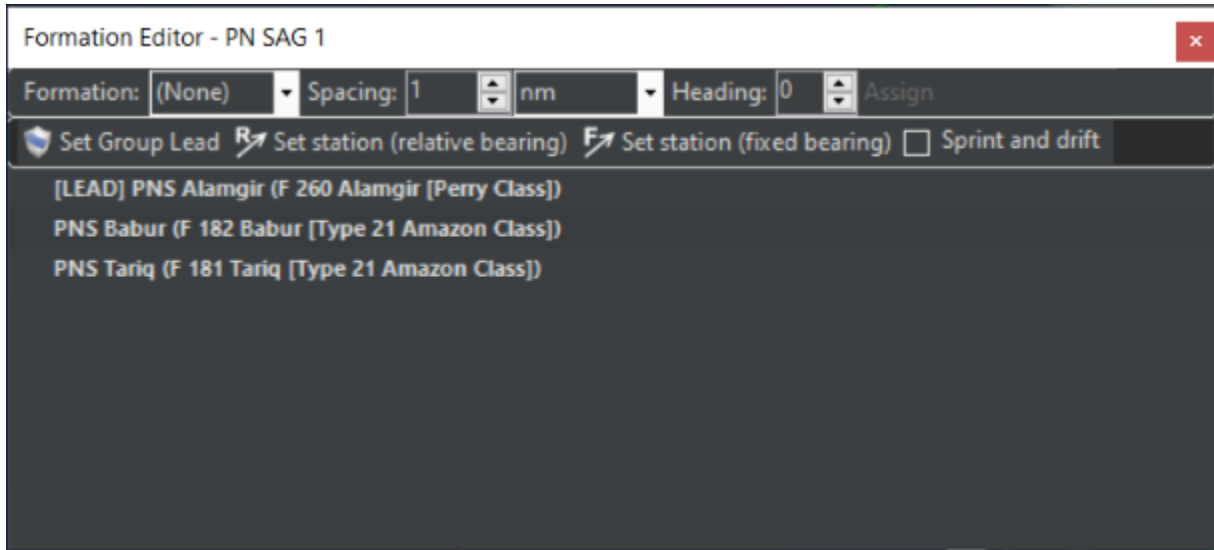
Special Case for Submarines: Submarines have special depth options for flexibility in being stealthy vs aggressive:

- **Just above the Layer:** The submarine will operate just above the Thermal Layer (See 9.2.3, Submarine Combat). This is optimal for use of towed array sonar.
- **Just Below the Layer:** The submarine will operate just below the Thermal Layer.
- **As deep as possible:** The submarine will operate at its test depth. If the sea floor is above test depth, the submarine will operate in a terrain hugging mode just above the bottom.



3.3.5 Formation Editor

This opens the formation editor to change the placement of units inside the group. You can also select this function by pressing the F4 hotkey. This opens the formation editor, allowing elements of a group to change position.



Formation: This sets the literal formation the units will form. It can be a line, column, wedge, or anything else on the menu. This menu allows you to select a custom preset formation. Formation presets are defined in the Resources/Formation folder in the main CMO directory.

Spacing: This sets the distance between units.

Heading: This sets the default heading between units.

Assign: This assigns the units to a formation, executing the changes.

Place: In the scenario editor, you can immediately order all units into a specific formation by pressing this button.

Group Lead: The lead platform is the center of the formation, usually a HVU (high value unit). When forming groups, the game will try and pick the best possible candidate but players can change if they desire to do so. The Group Lead is shown in the editor window with “[LEAD]” in front of its listing.

Setting Group Lead: Select the desired unit from the platform display drop down list and click the Set Group Lead button.

Setting Patrol Stations: Players can set patrol stations of units within the formation. To do so just select the unit from the formation editor listing so it is highlighted, click the set new station button (relative or fixed bearing) and then click a point on the map where you would like that position to be. You will know its created

when you see a diamond marking the position with the word station next to it on the display. The unit will then move towards it when the clock begins.

Set new Station (relative bearing): The assigned unit will always maintain its position relative to a group's course. This formation setting is most useful for ASW units, mine sweepers etc. as it clears a path for the HVU.

Set new Station (fixed bearing): The assigned unit will maintain its position relative to the lead regardless of course. This setting is most useful to put units between the HVU and a known threat.

Sprint and Drift: This allows units in the formation to, instead of traveling at a steady pace, achieve the same overall time by speeding up (sprinting) and slowing down (drifting). It is typically used by ASW units to get a better sonar picture by quieting their engines.

Note: Pay close attention to cavitation settings for ships using sprint and drift in a formation. Not being considerate of noise levels can give away your position without you even realizing it.

Note 2: A very powerful feature is the ability to manipulate formations through lua. This gives the ability to change formations based on location, threat axis, losses, etc. Below is a quick rundown on this capability.

Example of getting:

```
local f = ScenEdit_GetUnit( {name='Group 14', guid='GNRIJH-0HM7B6IC7IPUR'} )  
print(f.formation)
```

Example of setting:

```
local f = ScenEdit_GetUnit( {name='Group 14', guid='GNRIJH-0HM7B6IC7IPUR'} )  
f.formation = {name='wedge', transpose=true}
```

For setting the formation preset, the following options are available:

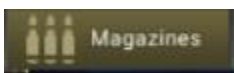
- name : Name of the preset as stored on StandFormations.txt
- transpose (optional): If True, the members are insta-teleported to assigned positions. If False (also default), the members are assigned their positions and will try to reach them as possible.
- spacing (optional): Override the preset's default spacing value
- spacing_unit (optional): Override the preset's default unit of spacing (0 for nautical miles, 1 for meters)
- bearing (optional): Override the group lead's desired heading

** General improvements on DB-Viewer DBID-Search*

- Navigation history for DBID searches
- Enter key handled
- Reset textbox on successful search

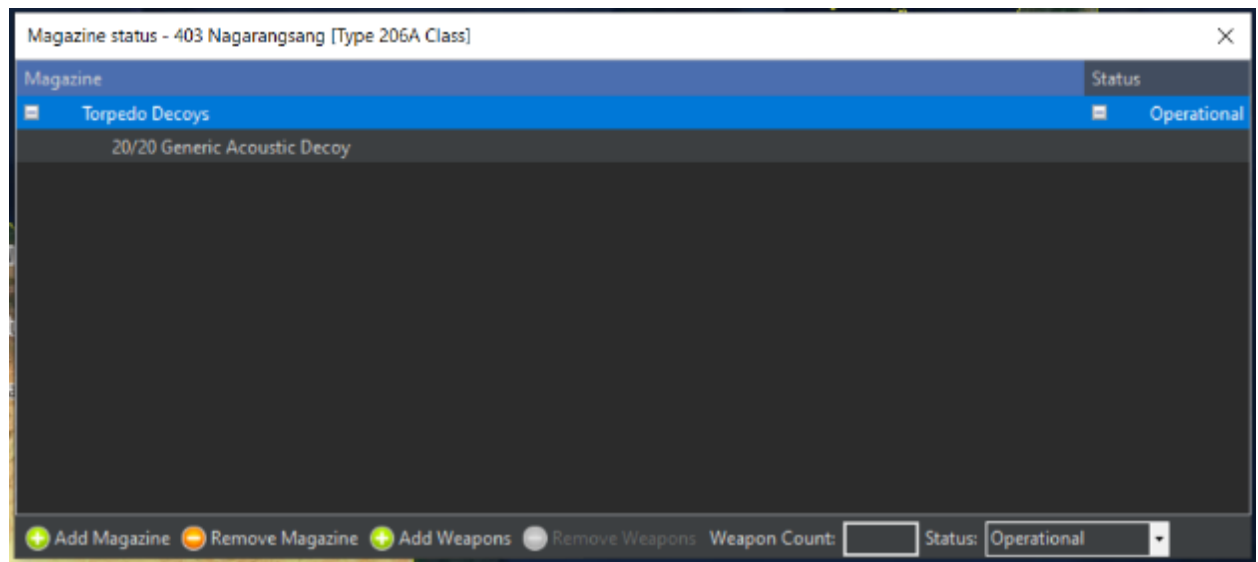
Refer to the lua documentation for more details.

3.3.6 Magazines



Pressing the magazine button launches the magazine dialog which displays individual magazines, weapons records and number of weapons carried.

In edit mode, you can add new weapons records to magazines and you can edit ammunition amounts in the weapons records. This allows you to supply mounts and hosted aircraft with the weapons and ammunition needed to any you have added to the host unit.

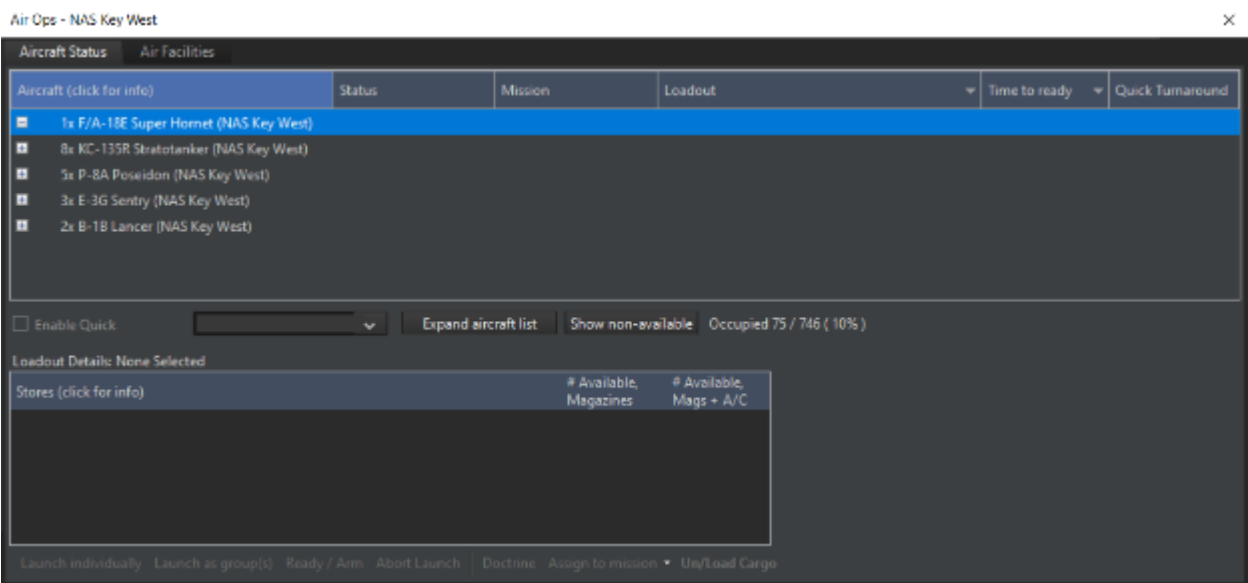


3.3.7 Air Operations



Clicking the button launches the Air Ops dialog which is used to manage hosted aircraft for the base in question.

The air ops dialog has two tabs: Aircraft Status and Air Facilities.



3.3.7.1 Aircraft Status Dialog

The aircraft status dialog groups aircraft by type with individual units shown in drop-down lists. You can expand and retract the lists by clicking the + and - button next to each unit type., The columns display current information on the hosted unit.

Aircraft: At the group level displays the unit count and type name. At the unit level displays the unit’s given callsign.

- **Status:** Shows the current status of the aircraft. These include:
 - Ready: A/C is available for use.
 - Readyng: A/C is in the process of reloading and refueling.
 - Taking off: A/C is in the takeoff process
- **Mission:** Shows the name of the mission the aircraft is assigned to. The field will be blank if unassigned.
- **Loadout:** Shows the current Loadout of the aircraft or the loadout being fitted if the aircraft is being readied.
- **Time to Ready:** Shows the time remaining until the aircraft is available. 0s is shown when the aircraft is ready. When an order is given for an aircraft to launch the launch time is then displayed. If quick turnaround is enabled, it will show the remaining quick-turn sorties.

- **Launch Individual button:** Pressing this button after selecting one or more aircraft will order them to launch and as individuals.
- **Launch as Group(s):** Pressing this button after selecting numerous aircraft will launch them as a group. If the selected aircraft are of different types and loadouts they will be launched as separate groups or individuals by type and loadout.
- **Ready/Arm:** Launches the Ready Aircraft Dialog which allows you to change the loadout of your aircraft with available munitions.
- **Enable Quick Turnaround:** In the editor, this enables or disables quick turnaround for the selected aircraft.
- **Expand Aircraft List:** This expands every available aircraft type to its full length.
- **Show/Hide Unavailable:** This either shows or hides aircraft that are currently unavailable for use.

The ready aircraft dialog displays a list of available loadouts and the ready time and attack altitude the primary munitions are deployed by the AI at. If you click on each row, the stores fields at the bottom will display the individual components (munitions, pods, tanks, sonobuoys) for each loadout and the number available in the parent units' aircraft magazine.

To choose one in the editor simply select the weapon row you like and press the OK-ready button or Ready Immediately (ScenEdit) button.

If the parent unit does not have the unit you can add it by clicking the check base magazine button at the top which allows you to view the magazine dialog and edit munitions.

Set time to Ready: You can set a value for when you'd like an aircraft to become available using this dialog. Just fill in the days, hours and minutes filed to complete. This is helpful for the start of scenarios.

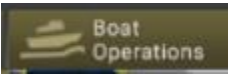


3.3.7.2 Air Facilities Tab

This dialog displays hosted aircraft grouped by the units and component facilities (hangars, open parking spots, revetments, elevators etc.). You can follow the progression of aircraft from hangar to elevator/taxiway to pad/runway. A launch time is also provided.

Facilities can be added or removed via the buttons at the bottom left. However, be aware that the unit must be flagged as “aviation capable” to be able to use fixed-wing aircraft.

3.3.8 Boat Operations



This function launches the Docking Ops dialog which allows players to manage embarked boats. This function can also be called with the F7 hotkey or clicking the boats button on the right hand information display.

Docking Ops - LS-1

Boat Status

Docking Facilities

Boat (click for DB info)	Damage (click for details)	Weapons (click for details)	Magazines (click for details)	Fuel	Mission	Status	Time to ready
2x LCM-6							
LCM-6 #3	No structural damage			100.0%	-	Docked	Ready
LCM-6 #4	No structural damage			100.0%	-	Docked	Ready

Collapse boat list

Occupied 34 / 50 (68%)

Launch individually

Launch as group(s)

Abort Launch

Doctrine

Assign to mission

Un/Load Cargo

The **Boat Status tab** displays docked boats grouped by type. The name at the group level gives a unit count and type name and at the unit level gives the individual unit name, status (locked, launching), assigned mission and time to ready or launch.

The launch individually button allows you to select one or more row and launch them as individuals. The launch as group allows you to select one or more rows and launch them as a group by type.

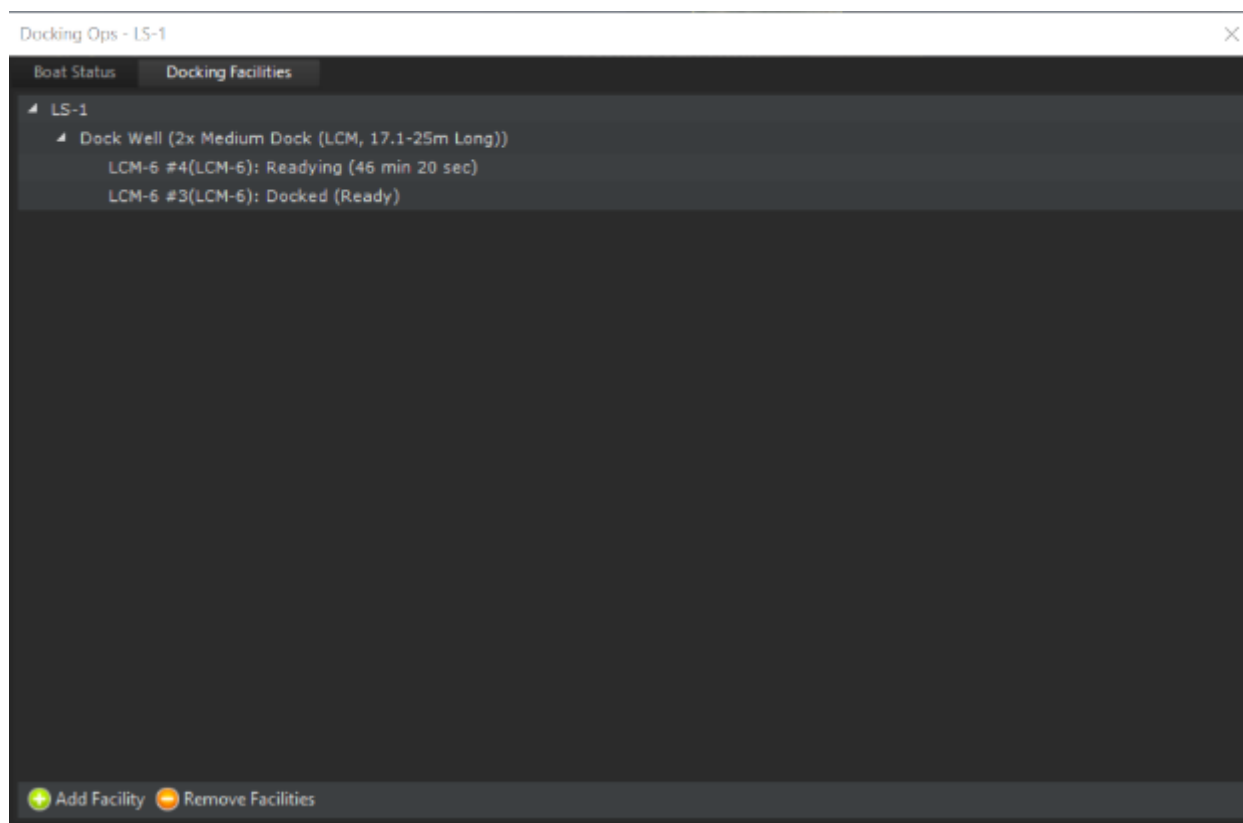
At piers, ships may be repaired and rearmed, although the process is extremely slow for all but the smallest ships. Watercraft may be returned to service with less than perfect health or ammunition from a pier. They can be deployed manually or automatically with the right doctrine (see 3.3.16, Withdraw and Redeploy).

Piers allow for more rapid and effective repairs than the jury-rigged ones capable at sea, and they can also load weapons more effectively than an UNREP.

However, piers are best suited for very long duration scenarios, not short ones where long-term damage management is not an issue. Thus, advanced port operations are not suited for a one or two day slugfest, but they are suitable for a long blockade or conflict lasting a week or more.

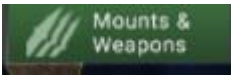
When placing piers or single-unit ports in the scenario editor, a blue area appears. Ships can move freely in this area even if it's otherwise overland. This is to let them reach the pier placed at a suitable location, and represents canals or channels made to allow entry to the port. Rotating the pier also changes the location of the blue area.

Multi-unit ports require ammunition storage facilities and piers like multi-unit airbases. Single-unit ports encompass all of these in one unit, also like their airbase counterparts.



The **Docking Facilities** tab shows the available facilities to hold smaller watercraft, similar to its counterparts at air bases.

3.3.9 Mounts and Weapons



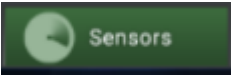
This function launches the Weapons Dialog which displays information on all weapon mounts on a platform. You can also launch it by pressing F8 on your keyboard or pressing the weapons button on the right-side information display.

Mount (click weapon for DB info)		Time to fire	Status	Reload Priority	Show Arcs
 20mm/70 Oerlikon	(15/15)	Ready	Operational		☒
15/15 20mm/70 Oerlikon Burst [20 rnds]	Gun			☐	
 20mm/70 Oerlikon	(15/15)	Ready	Operational		☒
 40mm/70 Single Bofors	(27/27)	Ready	Operational		☒
 Simbad [Mistral] Twin	(2/2)	Ready	Operational		☒

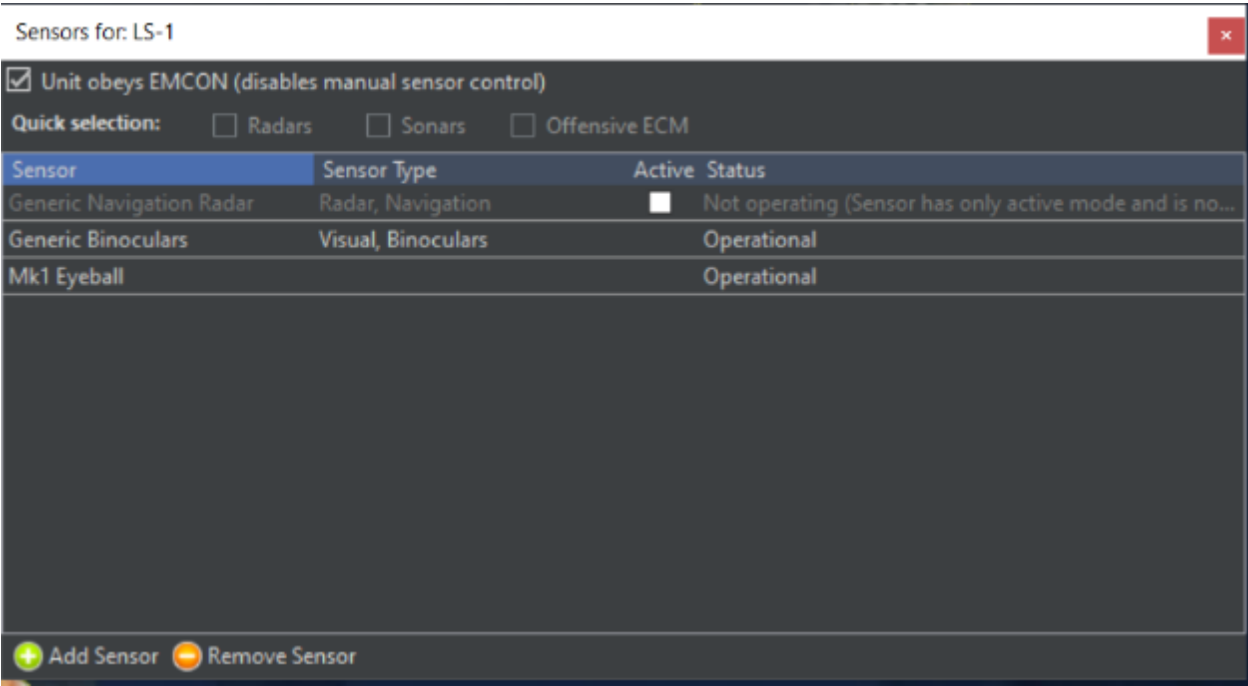
The weapons dialog displays a list of all weapons grouped by mount. The black root level text describes the mount, its capacity and status. The subset blue weapons record text describes the individual weapons in the mount, the time to fire and a checkbox that allows players to mark certain weapons as the reload priority in the case of a mount that shoots different weapons and ammunition. The gray subset text notes the capacity to carry the weapon but the lack of any.

- **Add Weapons Record button:** Allows players to add a weapons record to any mount by selecting the mount and pressing this button.
- **Remove Weapon Record(s) button:** Allows players to remove weapons records by selecting the record(s) and pressing this button.
- **Add Mount button:** Press this button to add a new mount.
- **Remove Mount button:** Select the mount(s) and press this button to remove them.
- **Edit Weapons:** To change the number of weapons in a record select the record, click the change button, enter a numeric value in the weapons field and then click set.

3.3.10 Sensors



This function launches the sensor dialog that displays a list of all sensors on a unit. This includes the sensor type, status and a check box showing whether the sensor is active or not. This function can also be activated by hitting the F9 key or pressing the sensors button on the display.

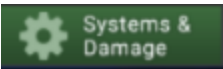


To make any changes you will need to uncheck the “**Unit obeys EMCON**” check box as EMCON setting are inherited from unit/mission or side settings. You can then activate or deactivate each sensor by either checking off the active check box in the list or activating them in groups by using the Quick selection check boxes. Keep in mind that passive sensors are always on and cannot be turned off.

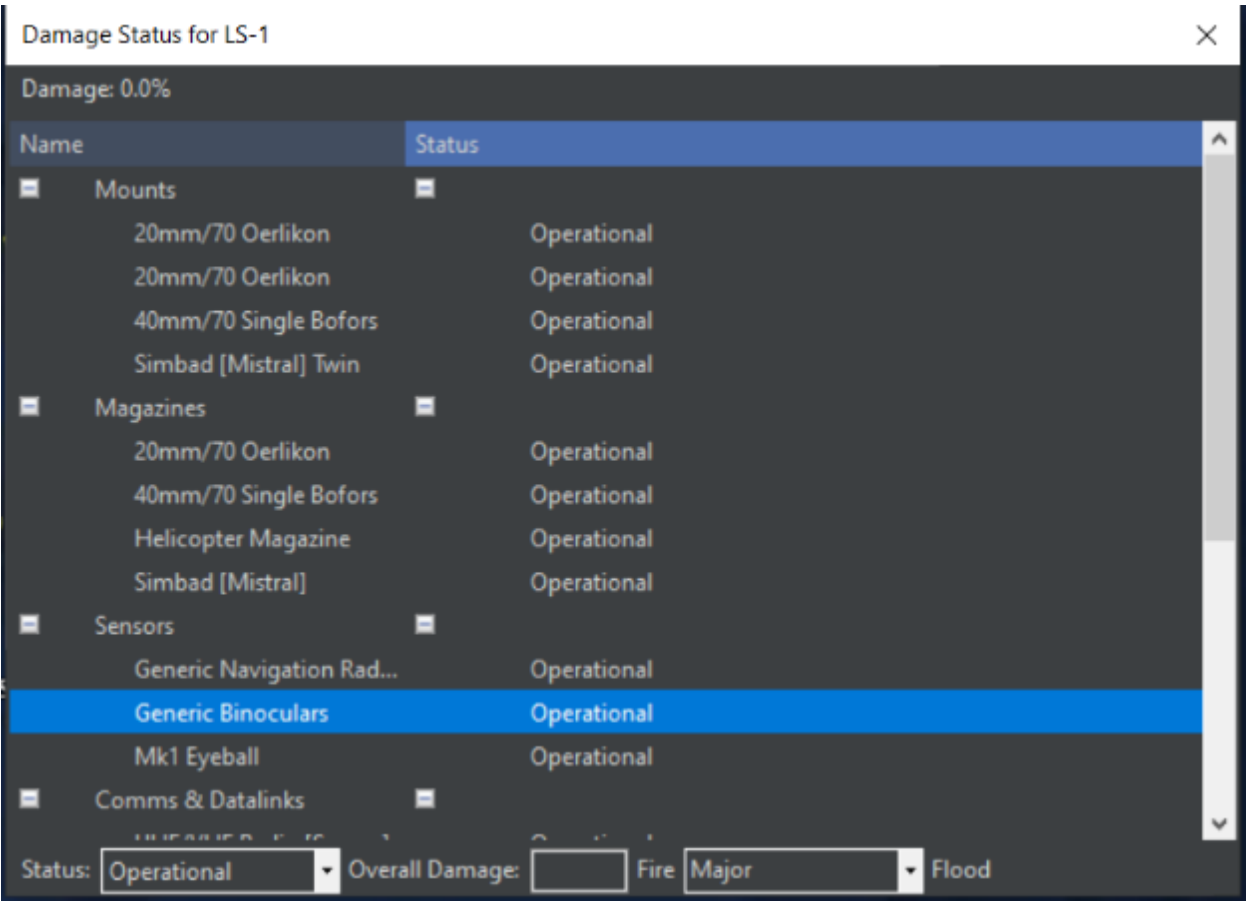
- The **Add Sensor** button launches the **Add Sensor Dialog** which allows you to add a sensor to a platform.
- The **Remove Sensor** button works by selecting a sensor and then pressing this button.
- **Damage Control:** This function opens the Damage Status dialog (F10 hotkey) and “Damage Cntrl” button in the Right Side Information Panel as well. This displays the damage conditions of the unit and its sensors, mounts and magazines.

In **Editor mode**, you can change the values of these to add damage to ships and units in constructed scenarios.

3.3.11 Systems and Damage

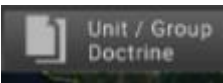


Opens the Damage Status window. This displays the damage conditions of the unit and its sensors, mounts and magazines.



In Editor mode, you can change the values of these to add damage to ships and units in constructed scenarios. You can also light the unit on fire.

3.3.12 Unit/Group Doctrine



This allows the doctrine, EMCON (Emissions Control, aka whether the active sensors will be on or off), and Weapon Release Authority of the selected unit to be changed. It can be changed side-wide under the “**Game Menu**”, the “**Edit Sides**” function in ScenEdit, or by pressing CTRL+Shift+F9.

The doctrine window has four tabs: General, EMCON Settings, Weapon Release Authorization (WRA), and Withdraw & Redeploy.

Note: With General Doctrine, EMCOM, Weapons Release Authorization (WRA), and Withdrawal & Redeploy settings, you are given an immense amount of control over how your units, groups, and missions behave without the player having to intervene. Mastering these settings is key to mastering Command.

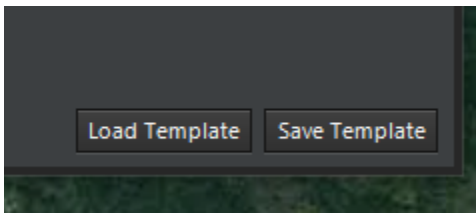
General Info on The Doctrine windows

Doctrine and RoE options under the General tab include the following:

All doctrine options have the “inherited” setting. This means the unit automatically follows the doctrine of the side (if unassigned to a mission) or a mission (if assigned, and the mission’s ROE/Doctrine contradicts the side). The buttons at the bottom of the “doctrine window” can be used to reset doctrine to the default inherited settings of the side if need be.

In the scenario editor, all doctrine settings can be set to play-editable through a checkbox to the right of the individual doctrines. For example, a scenario editor can let a player change the refueling threshold of force the player to use a general threshold of 30%.

At the bottom right corner of the General Doctrine page are the options to save and load all the WRA and doctrine information as a file that can be easily loaded at a later date.



STRATEGIC:

Use Nuclear Weapons: Determines if the platform is authorized to employ nuclear weapons. If enabled, it will fire them. If disabled, it will not.

RULES OF ENGAGEMENT:

Weapons Control Status: Determines rules of engagement for either air, surface, subsurface, or land targets. Weapons TIGHT means the unit(s) will fire at confirmed hostile targets only. Weapons FREE means it will fire at anything that isn’t confirmed friendly (possibly dangerous to neutrals or allies). Weapons HOLD means it will not fire on its own at all and will launch only if manually ordered.

Ignore Plotted Course: If enabled, the unit will abandon its previously (manually) plotted course while attacking a target. If disabled, it will not.

Engage Ambiguous: Sets the AI behavior when considering using a weapon. Each weapon has certain tolerances calculated on-the-fly for down/cross-range; most direct-guidance systems want a precise target, most LOAL weapons have relatively strict cross-range tolerance (equal to the sensor swath) but relaxed downrange tolerance because of their BOL ability, area-effect weapons can tolerate up to their kill zone etc. If

the target's ambiguity figures are greater than the weapon tolerances, this is explained in the manual weapon allocation window and the weapon will not be fired. This dialog has three options besides "inherited".

An ambiguous contact is any contact with an uncertainty area. When set to "ignore ambiguity", the AI will ignore the ambiguity when determining whether or not to launch a weapon. Select this at one's risk, for the chance of hitting a phantom or moved contact is incredibly high. When set to "Optimistic", the target uncertainty area must be smaller than 3x the weapon tolerance for the AI to launch the weapon. When set to "Pessimistic", the target location uncertainty must be less than the weapon tolerance for the AI to launch the weapon.

Engage Opportunity: If enabled, the unit will engage targets not directly related to the mission at hand. If disabled, it will not. Enable at one's own risk, for this can lead to dangerous behavior.

Emission Control (EMCON)

Ignore EMCON when under attack: If enabled, the unit will turn on its emitting sensors when it's actively engaged defensively (the theory being that since it's already been found, there's no use hiding and weakening its defenses anymore). If disabled, it will continue with its previous EMCON state even if attacked.

MISC:

Kinematic Range: If enabled, the unit will fire torpedoes to its kinematic (maximum physical) range. The much shorter "practical range" is there to ensure that the target cannot avoid the torpedo simply by running at flank speed in the opposite direction. Kinematic range can be set for always practical, always kinematic, or kinematic for manual shots. The kinematic for manual setting means a submarine set on a patrol mission running into warships will engage at a practical range, but if the player sees a slow merchant ship with no sonar, they can engage it from a distance with a manual allocation.

Automatic Evasion: The game has several pre-programmed evasions routines that the AI will undertake if it detects it is under attack. This RoE gives you the ability to turn this behavior of and off. In some cases, turning it to NO is useful if having aircraft pressing their attacks is desired.

Refuel:

Refuel/UNREP: This changes whether or not aircraft will refuel and ships will resupply each other. It can be turned on, turned off, or (default) turned on but not have tankers attempt to refuel each other.

Refuel/UNREP Selection: This allows tankers to be chosen, be they simply the nearest ones, tankers closer to the objective, or an even greater priority to closer-target tankers.

Refuel/UNREP Allied Units: This allows supply units to refuel ones from sides other than the player. Available choices are allowing it, allowing either onloading or offloading only, or forbidding it altogether.

Refuel to Tanker Threshold: Sets the threshold for when an aircraft will seek out a tanker if set to refuel.

Air Operations:

Air Operations Tempo: This changes the overall tempo of the side's air operations. A “**surge**” tempo generates sorties at a far quicker rate than a “**sustained**” tempo. Historically, “**surge**” tempos cannot be maintained for very long. Sustained is thus recommended for a scenario set in the middle of a long air campaign, at the end of a long logistical chain, or both. (There were two “Surge” periods in the Gulf War air campaign, at the beginning and during the ground war.)

Note that tempo is different from the ready times included in the aircraft loadouts.

Quick Turnaround: This enables quick turnaround for suitable aircraft and loadouts.

Quick turnaround is used to model frenzied air activity such as the Israeli strikes at the beginning of the Six Day War. It enables aircraft to repeatedly fly brief missions with a short ready time before the inevitable cooldown.

- **YES**-all possible quick turn platforms are enabled.
- **FIGHTERS AND ASW**-This enables non air-to-ground loadouts for quick turnaround. Air to ground ones are disabled and ready at the same overall tempo.
- **NO**-No quick turnaround allowed. Loadouts ready at the default operational tempo.

Fuel State/Pre-Planned: This changes the time when an aircraft automatically considers itself having insufficient fuel and returns to base. The default is “Bingo” (just enough to make it back), but can be changed to various “Joker” states to provide an additional barrier against running out of fuel.

Fuel State/RTB: This changes when the individual aircraft in a flight return to base. It can be set to individual units leaving, when the first aircraft reaches bingo/joker (encouraged), when the last aircraft does so (EXTREMELY risky), or not RTBing at all at the fuel state (Even riskier yet, only done in the event of some assured micromanagement).

NOTE: The default fuel RTB settings are there for a reason. Change at your own risk and be aware that losing the aircraft to a lack of fuel is possible, even likely.

Weapons State/Pre-Planned: This changes the time when an aircraft automatically considers itself “Winchester” (out of suitable weapons) and returns to base. The aircraft can be used for anything from one BVR launch and run engagements to fighting to the last shell, depending on the situation. Fuel retreat can also be modified.

- **Use Loadout Setting:** The default, this goes with the inherent loadout.

For others, an example is in order: An F-14 Tomcat with a mixed arsenal of AIM-54s, AIM-7s, AIM-9 Sidewinders, and its gun takes off for a patrol mission.

- **Winchester:** This disengages weapons concerned with that mission. In the event of an air to air mission, it means everything that can fire at aircraft. In the event of opportunity fire being set, it will engage with air to air guns. In this situation, the Tomcat will fight until it runs out of ammunition (or gets low on fuel). If it's serving as a screen for ground units against enemy attackers or bodyguard for a high-value unit, this setting is effective.
- **Shotgun:** Less than the maximum number of weapons will be fired before the unit RTB

- **All BVR or Standoff** means that the hypothetical Tomcat will fire both its AIM-54s AND AIM-7s before returning to base, useful if engaging say, a bomber formation.
- **One Engagement with BVR or Stand-off** is useful if it's facing a superior opponent. It will engage once with (presumably AIM-54s) BVR missiles, and then RTB. That way it can take advantage of range in a hit-and-run engagement.
- **One Engagement with "WVR or Strike"** is useful if it's a ground attack aircraft you don't want hanging around in the face of AA weapons. It's also useful for a similar swoop-and-run attack done with an aircraft that only has WVR weapons.

Make sure to turn targets of opportunity off if you don't want strange actions like an MPA or bomber flying over to engage a comparable enemy plane with its tail guns. (*British Nimrods were armed with AIM-9s in the Falklands so this sometimes happens!*)

Weapons State/RTB: Like with the fuel state, this changes when the individual aircraft in a flight return to base. The "last aircraft reaches _____ state" options are considerably less risky than with fuel.

A/G Strafing (Gun): This enables or disables strafing gun runs on air-to-ground missions. Whether it is to be enabled or disabled depends highly on the circumstances-a "fight to the death for the sake of ground support" situation with an inherently expendable airplane that has only a small number of bombs would support it being enabled, while an advanced aircraft operating against a target with a high level of AAA would support it being disabled.

Jettison Ordnance: This enables or disables an aircraft jettisoning its A2G ordnance when under attack. An aircraft with no ordnance will be more maneuverable thanks to being lighter, but it also obviously cannot perform its ground attack mission. This was the real goal for North Vietnamese interceptors-shooting down American aircraft was just a bonus luxury.

BVR Engagement Logic: This changes the doctrine aircraft will follow when engaging with long-range missiles.

- **Follow Missile Straight-in:** The aircraft will continue to follow the missile straight towards the foe. This can be used to simulate a unit with rigid doctrine, poor training, or a desire to close to visual range quickly.
- **Crank if Possible (DEFAULT):** The aircraft will "crank" by tilting to the edge of its guidance radar and slowing down. This symbolizes a desire and training to stay at long range if possible.
- **Crank and Drag:** The aircraft will not only crank but will, once the missile goes active and its guidance is no longer necessary, "drag" and move away from the target. This symbolizes a greater desire and skill to stay at long range at any cost.

Anti-Surface Warfare:

Use SAMs in ASuW Mode: If enabled, ships with SAMs that can fire them in anti-surface mode (which most radar-guided SAMs can achieve) will fire them at appropriate surface targets.

Maintain Standoff: If enabled, units will attempt to stay inside the range of their own weapons but outside that of their known opponents. If disabled, they will charge the target and attempt to engage with every available weapon (which can range from a destroyer firing its missiles and closing to gun range knowing the danger, or it can be a missile boat with only a machine gun on deck trying to engage with that).

Navigation: Select the appropriate method to navigate between waypoints.

Use missile waypoints: When set to Yes, ASuW missile shooters will use the missile's waypoint/dogleg capability (if present and if the range allows it). If set to No, ASuW missile shots will always go "straight in".

Submarine & ASW Warfare:

Avoid Contact: If enabled, the submarine will attempt to avoid contact with any known threat. "Yes-Always" makes it non-negotiable, while "Yes-Self Defence" means it will at least fire back to defend itself.

Dive When Threat is Detected: If enabled, the submarine will automatically dive deeper when it detects a threat within the exact parameters of the setting. This can be a matter of radar detection, ESM detection, or simple proximity.

Recharge Battery %, Transit/Station: This determines the threshold when a submarine will rise to recharge its battery and is not in an urgent engagement.

Recharge Battery %, Offensive/Defensive: This determines the threshold when a submarine will rise to recharge its battery and is in an urgent engagement.

Air-Independent Propulsion: This determines whether or not a submarine with air-independent propulsion capability will use it. Options are "always", "never", and "when engaged offensive or defensive" (aka, in urgent battles).

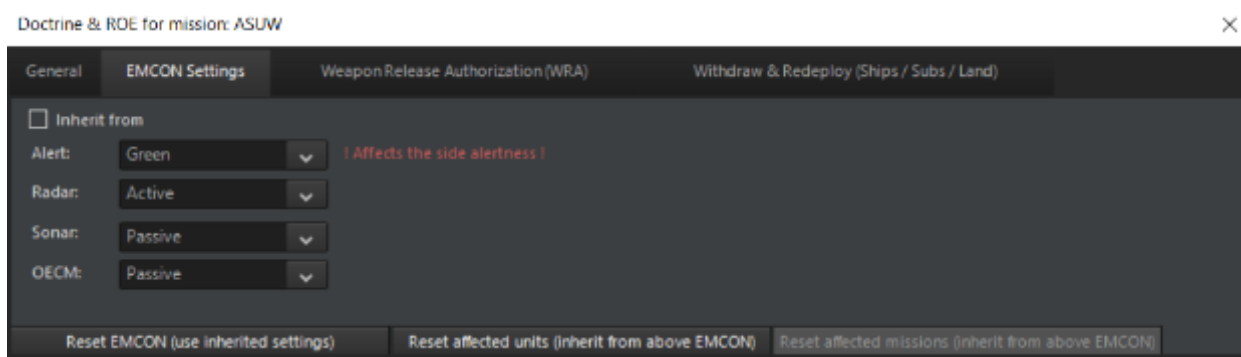
Dipping Sonar: If enabled, helicopters will automatically deploy dipping sonar when hovering at a low enough altitude over water. If disabled, they will only do so if on a mission or manually ordered.

Land Warfare:

Navigation: This determines the type of pathfinding a land unit takes. If set to "shortest route" (the previous default) the unit will attempt to steer on the easiest path. If set to "direct route", it will simply move in a straight line towards its destination without any consideration for the terrain types.

(WARNING: This may result in the unit getting "caught" on difficult terrain such as mountains. Use at your own risk)

EMCON Tab



EMCON is the Emissions Condition of the platform/side/mission. It allows radar, active jammers, and active sonars to emit by default or not.

This sets the ordered status of all emitters regarding Radar, Active Sonar, and OECH (Offensive Radar Jamming). Each emitter category has its own setting and has two states:

- **PASSIVE:** Radars, Active Sonar, and OECH off.
- **ACTIVE:** Radars, Active Sonar, and OECH on. Be aware that active sensors can be detected at greater distances than their sight range (think the equivalent of seeing a flashlight beam) and should thus be turned on only when necessary.

Setting EMCON: There are several ways of setting EMCON for a unit or group of units. At the beginning of the game, the EMCON settings have been set by the scenario designer. If the player does nothing, all Platforms will inherit the initial EMCON settings. The player can manually set EMCON settings in a variety of ways.

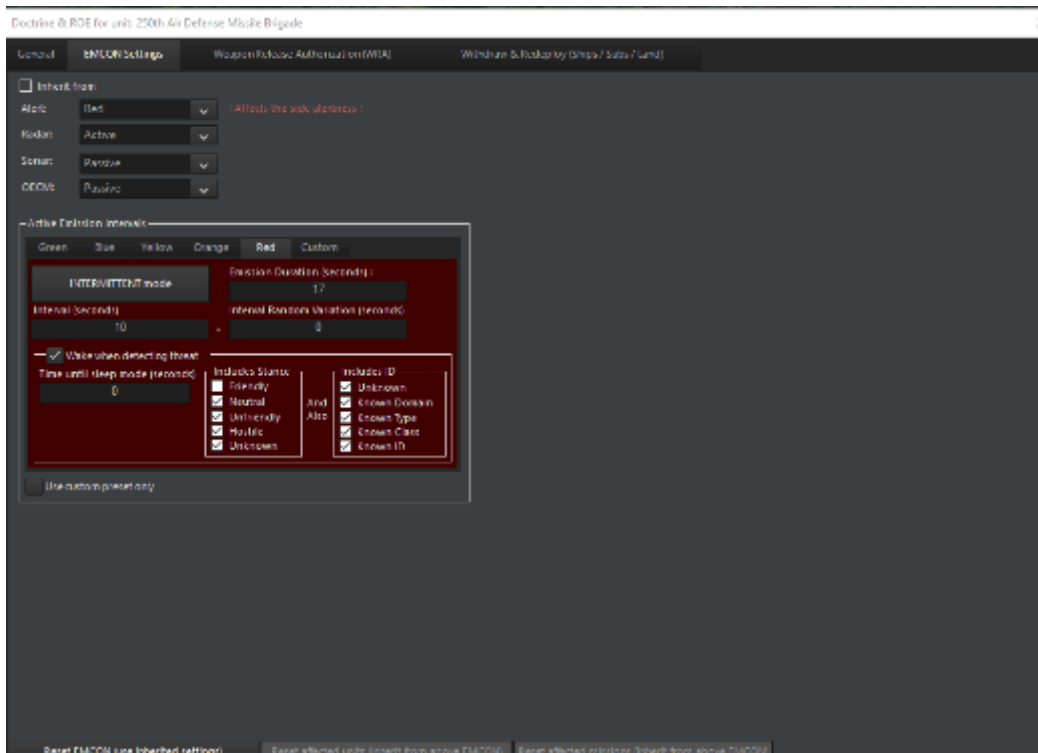
EMCON Default Value: The default value for EMCON is "Inherited", where the platform inherits its EMCON condition from its parent platform and/or as set for that side by the scenario designer.

Inherited EMCON: EMCON can be inherited from the scenario designer settings, Mission EMCON settings, Group EMCON settings, or Parent Unit EMCON settings. Platforms assigned to a mission will inherit the mission's EMCON settings and any embarked or assigned platforms will also inherit them. Aircraft automatically inherit their EMCON settings from either their parent platform or their mission is assigned to one.

Manual EMCON: setting the EMCON settings of a platform or a group. Click on the unit or group and select the "SENSORS" button on the unit information window. Set the desired EMCON settings. You may get a message to confirm the manual override of the EMCON settings.

Intermittent EMCON: Different "alert" levels can be set with sensors turning on and off at either random or set intervals, or in response to threats being detected. This can be useful for allowing active sensors to be used while not constantly blaring them, or for adjusting them based on context.

ALERT LEVEL (side): The Alert allows a player to see and modify the side-wide alert level. Alert levels are Green, Blue, Orange, Yellow, and Red. The color designation has no meaning in reference to other colors. It only has meaning for how you configure the EMCON for individual groups and units on that side. The five colors allow the player to configure five alert levels for a side. The alert level can be changed manually by the player or changed based on events and lua code.



ALERT LEVEL (unit or group): The picture above is of the EMCOM display for a unit or group that can be accessed through the menu, right clicking, or the buttons along the top of the map. While the sensors can be controlled basically the same as the Side EMCOM described above, it also includes an Active Emissions Intervals box. This is where Intermittent Emissions are configured.

Different “alert” levels can be set with sensors turning on and off at either random or set intervals (Intermittent Emissions), or in response to threats being detected. Intermittent Emissions can be useful for allowing active sensors to be used while not constantly blaring them, or for adjusting them based on context. The Active Emissions Intervals box is where the emissions setting are input for each alert level plus a custom setting. The custom setting allows the player to have more granular control of intermittent emissions that aren’t tied to the alert level. Keep in mind that for intermittent emissions to work, the sensor must be set to active, not passive.

Active Emissions Interval box label definitions:

Green/Blue/Yellow/Orange/Red/Custom tabs – Separate intermittent emissions configuration screens for each alert level plus a custom level.

Intermittent/Continuous Mode – Switches between the Active Emissions Interval box and the alert level controlling emissions and passive/active drop-down boxes controlling emissions. If it says Intermittent mode, it is in intermittent mode. And the same for Continuous mode.

Emission Duration – How long, in seconds, the emissions will be turned on.

Interval – How long the emissions will be off, in seconds, before they cycle back on.

Interval Random Variation – A random variation added to the interval off cycle. Will be no longer than amount entered in seconds.

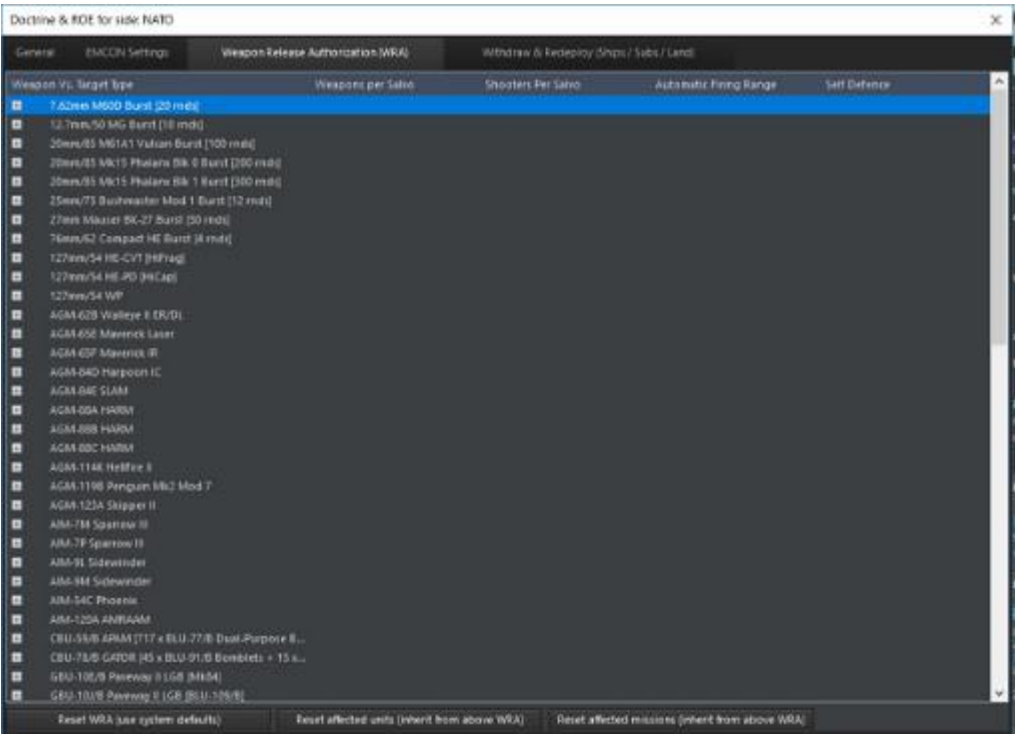
Wake when detecting threat – Will force the sensor to power on if a new unit is detected. If the player leaves Intermittent mode on, the sensor will stay on until the enemy unit disappears and the sleep mode expires

Time until sleep mode – How many seconds will have to count off after a detected unit disappears for the sensor to go back into its intermittent cycle.

Includes Stance and Includes ID - As you can see, the wake mode can be configured to activate with filters from detected unit posture and detection level.

Warning: The Alert level setting is for the entire side. If you change the alert level in a unit’s EMCON screen, it changes the entire sides alert level. If you want intermittent emissions for units and groups to not be impacted by alert level, use the custom tab in the Active Emissions Interval box and make sure the Use Custom Preset Only box is checked.

WRA Tab



Weapon Release Authorization allows players to tailor different weapons for different targets. Unknown/unspecified is for targets not yet positively classified. Thus, an aircraft can be set to fire two AMRAAMs at a time at a fighter target and one at a time against a wallowing support aircraft.

WRA refers to the target’s “Missile Defense Value”. This is included in the target’s database viewer and is a guide to ideally how well they can defend themselves against a basic Harpoon-style ASM. A small missile boat or unarmed civilian ship has a missile defense value of 2 Harpoon-equivalents, while a high-end version of the Ticonderoga or Kirov cruisers has a whopping 96. Note that missile defense value only applies to positively identified targets.

For warships, WRA is extremely useful to ensure that heavy missiles are not squandered on overkill targets (ie, don't launch monstrous carrier-busting missiles against small sailboats).

WRA also allows for maximum firing range to be changed to a distance less than the on-paper maximum. This is especially useful for scenario designers to ensure a higher probability of kill for their AI side, as the missile will have plenty of energy. It is also helpful to score kills against alerted enemy aircraft without wasting the missiles, as they will dive to avoid attacks, dropping below the horizon and causing direct-line-of-sight guided missiles to lose their lock.

Another function of WRA is giving the unit "personality". On one extreme, a jumpy, ill-trained crew can be portrayed by setting the WRA to engage at the farthest range possible and fire the most weapons possible, regardless of the target. On the other, an overly cautious crew can be set to only engage at very short ranges and only launch one weapon at a time. A balanced unit will be in the middle.

WRA can be combined with other doctrine and scenario measures to further flesh out the unit personality. These include side proficiency and engaging non-hostile targets (the jumpy unit should engage them the moment they appear, the cautious one will wait until they are 100% sure it's hostile.)

The WRA can be set to a no-escape zone launch that causes a weapon system to only engage when the current target cannot 'escape' the weapon using a course reversal maneuver. If the target contact is unknown, the weapon will be launched at it using this option based on the unknown target's current speed. If the target type is known, the weapon will launch based on the target's *maximum speed*. This can result in shots waiting for a last possible moment in the instance of a very high-speed target.

A final instance of unit personality is in emissions control. Many otherwise-skilled units have proven sloppy or clumsy with emissions secrecy in wars throughout history, and this can be simulated by having radars set to "ON" by default even when it is not opportune.

Here are some typical examples of "personality" for aircraft are, using the same fourth-gen fighter with radar-guided missiles as a base:

- **Default.** RTB weapons state depends on loadout, WRA is two radar guided missiles (AMRAAM, etc) at a time against a known fourth- or fifth-generation fighter and one against anything else. This is a sensible choice for most circumstances.
- **Hit-and-run.** RTB weapons state is Shotgun, one engagement with BVR, WRA dumps all the BVR missiles at the target, with WVR ones possibly disabled altogether or at least toned down. (*This can be a maddening adversary if he out-ranges and outruns you*)
- **Fight to the death.** RTB weapons state is Winchester, all mission specific weapons expended, allow targets of opportunity with guns. WRA close to default. This can be taken to extremes by simply selecting the "No, aircraft do not RTB when weapon state is reached" button.

Sometimes, different missions call for different personality sets. An interceptor operating in an environment of enemy air superiority would be suited for a hit-and-run attack, like North Vietnamese MiG-21s which near-invariably fired their AA-2s and then ran for home. An escort fighter would be more willing to fight to the end to

make sure the enemy goes for them and not their charges. A nervous or inexperienced pilot will fire all their missiles at near-max range, while a veteran will wait until they can engage in optimal conditions.

(Those optimal conditions vary by weapons and target, and should be found by scenario editor experimentation.)

EMCON is up to the player to interpret-it's possible to have a pilot/crew who cluelessly exposes the radar, or a timid one that never turns it on even when it would be justified. Both show inexperience and a lack of potential in their own ways.

And of course, mission setups, especially for AI players, illustrate the overall personality of the *side*. A timid aircraft may launch and run, but a timid *air force* will simply stay on the ground.

Withdraw/Redeploy Tab

Doctrine & ROE for side: Italy ×

General EMCON Settings Weapon Release Authorization (WRA) **Withdraw & Redeploy (Ships / Subs / Land)**

Withdraw when....

Damage is more than:

OR Fuel is less than:

OR the primary attack weapon is at less than:

OR the primary defence weapon is at less than:

Redeploy when....

Damage is less than:

AND Fuel is at least at:

AND the primary attack weapon is at least at:

AND the primary defence weapon is at least at:

This sets the doctrine for ships assigned to a port unit (see above, Docking Operations).

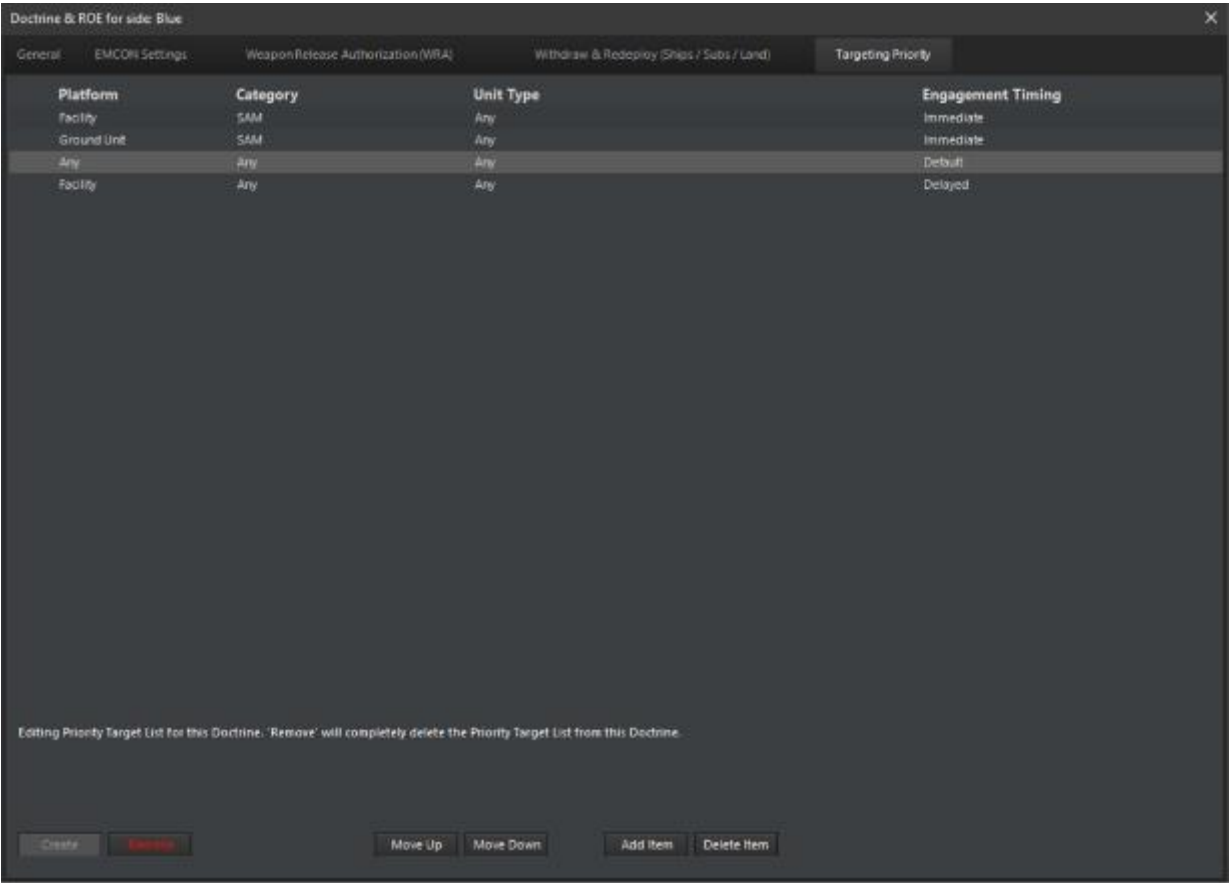
Ships can be configured to automatically withdraw from the engagement and return to the nearest port if they are either damaged beyond a certain degree or low on fuel or ammunition. This is necessary for 'intelligent' AI missions and can be useful for a human player as well.

The default is set to "ignore", meaning that ships will stay on their mission no matter what.

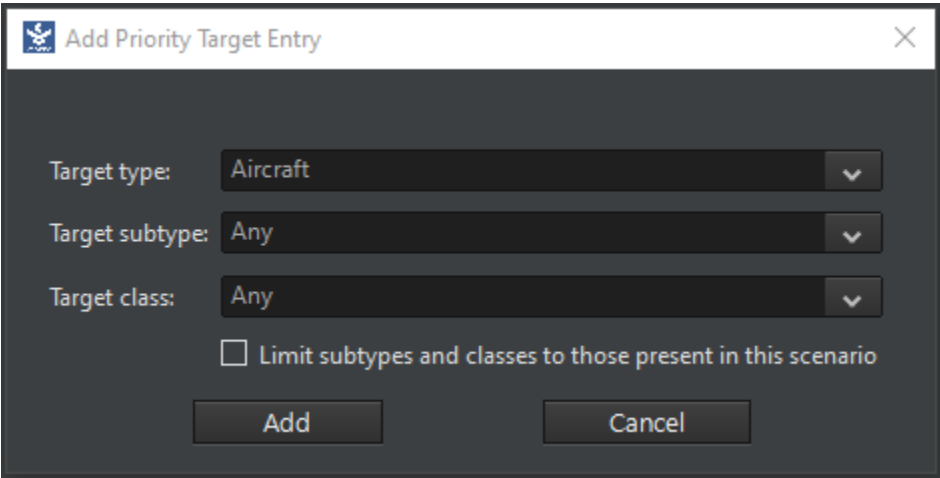
The redeployment settings control when a ship is to re-emerge from port. The defaults have them only setting sail again at full fuel and ammunition loads less than 5% damage. The changes can simulate an emergency redeployment, or one where a full load merely isn't necessary.

Targeting Priority

The fourth tab on the Doctrine & ROE menu is the targeting priority page. This allows you to customize what the priority targets are for a given side, mission, or unit. To create a new set of priorities, select the create button in the bottom left corner and then click on Add Item to create a new priority for the unit.



Automatic and manual attacks will ignore the priorities set here, so this tool is be used for units assigned to pre-planned missions and units that engage targets of opportunity. After pressing Add Item, a new dialog will appear:

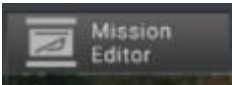


This menu has several options to customize your priority.

- **Target Type:** This picks the family or unit type of targets. It can be left broad, or clarified with the other options.
- **Targeting Subtype:** This is the specific subtype within a type of targets that is focused upon with this prioritization. For example, if you used Aircraft as your Type, the subtype could be bombers.
- **Target class:** This lets you pick the exact unit you wish to engage with this priority. If the previous subtype was bomber, this would allow you select the exact DBID bomber you wish to engage.
- **Limit subtypes and classes:** This checkbox allows you to only see units in the scenario as options to select.

After creating several different prioritizations, you can now order them in the main menu by pressing the Move Up and Move Down options. The target prioritization is affected by the WRA, so keep in mind that you may create a situation where multiple units must have enough 'weapons on the way' to a specific priority target before engaging down the list of targets.

MISSION EDITOR

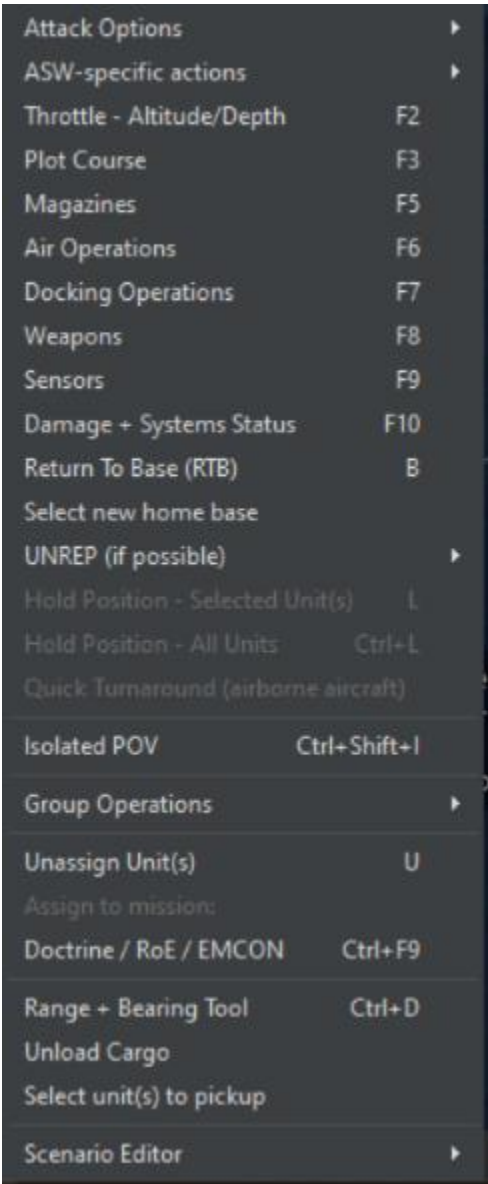


The mission editor is the backbone of COMMAND, and in addition to the shortcut bar, can be accessed by either pressing F11 or using the "Missions + Ref Points" drop down menu. The complexity of the mission editor is such that it deserves its own section. See 7.2, Missions, for in-depth information on the types of missions.

4.0 Menus and Dialogs

4.1 Right Click on Unit/ Context Dialog

You can right-click on a unit to display a drop-down menu of popular functions in the game.



4.1.1 Attack Options

Engage Target(s) - Auto: This function will order the selected unit or group to attack a designated target or group of targets. See “Engage Targets- Auto” under the “Buttons and Windows” section for more information.

Engage Target(s) - Manual: This function calls on the weapons allocation dialog, which allows players to manually allocate weapons to selected targets. See “Engage Targets- Manual” under the “Buttons and Windows” section for more information.

Bearing-Only Attack: The bearing only attack allows players to fire guided weapons without designating a target but instead designating a point which their guiding sensor(s) will activate to hunt targets. It otherwise functions identically to the “engage target(s)-manual” function.

Be aware that doing this may put neutrals at risk as the sensor may not know friend from foe and that it has a high chance of missing, especially if the contact location is not accurate. You can also call on this function using Ctrl+F1 hotkey combo or from the Unit Orders drop down menu.

(For purely inertial-guided weapons, like ballistic missiles or JDAM-type bombs, the designated point is not the seeker-activation point but rather the desired impact point. This can be useful if, for example, you want to finetune the airburst point of a cluster warhead.)

Lay Chaff: This function has the aircraft deploy a cloud of anti-radar chaff. The “single burst” option releases one ‘puff’, while ‘continuous stream’ has the aircraft deploy chaff until it runs out or has to RTB for another reason. The corridor and its altitude are marked on the map.

Aircraft inside the corridor have an effectively reduced RCS (the chaff strips break up the radar beam and scatter it wildly), but it is dependent on them staying inside it. Improvements in radar quality made chaff increasingly less effective in real life, as aircraft would be confined to a narrow corridor with increasingly marginal benefits from it.

Note that the aircraft must be *inside* the corridor (and at the same altitude as the cloud when faced with 3D radars), not merely behind it. Chaff-droppers must therefore precede the main body, an inherently dangerous task (many F-4 crews assigned to such a role in Vietnam found this out the hard way.)

From the late 1970s onward, chaff sharply drops in effectiveness. 3D and phased-array systems, along with better electronics overall, can make radars more capable at seeing through it. Experiment in the editor to see what radars can and can't be foiled by chaff.

Drop Target: This function allows a player to drop one or more units from its target list. After selecting this function simply click on the unit you would like to drop or drag select multiples.

Disengage (Drop All Targets): This function drops all targets from the unit target list and the unit will continue with its previously assigned mission or plot.

Ignore Plotted Course When Attacking: If units are on plotted courses and are given an attack order they will fly to the end of their plotted course and carry out the attack by default. Selecting this option turns that behavior off and units will attack immediately.

Weapon Control Status (WCS): This allows the weapon control status (see “General Tab” below) to be changed. WCS Selected Unit(s) changes the settings only for the units in question, while “All Units” changes it for the entire side.

4.1.2 ASW-specific Actions:

Drop Passive Sonobuoy: If the aircraft has sonobuoys in its inventory, it drops it. Sonobuoys can be dropped above or below the thermal layer (see 9.2.3, Submarine Combat). Passive sonobuoys are the main source of anti-submarine warfare.

Drop Active Sonobuoy: This drops a sonobuoy that actively pings. As the submarine will know it's being watched, active sonobuoys are traditionally used only to pinpoint the exact location of an already-detected contact prior to an attack.

Deploy Dipping Sonar: If the aircraft is a helicopter with a dipping sonar, it stops, hovers low, and deploys it.

Function key functions: Allow you to call on unit level functions. See the shortcut bar, "Buttons and Windows" for more information.

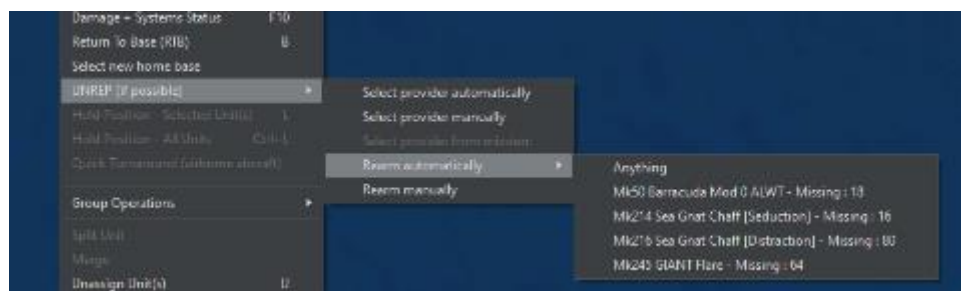
4.1.3 Context Menu, Cont.

The Range and Bearing Tool: This tool acts as a measuring tape for players. To use it select the option (or hit Ctrl + D) and click an point on a map and drag out you will see it measure range in nautical miles and bearing relative to the point you chose.

RTB: If an aircraft is selected using this function will order an aircraft to return to its home base.

Select new home base: If you'd like to designate a new home base for a unit select this option and click on the new home base or ship on the map. Home bases can be ports (for ships), airfields (for aircraft), or ships with the size and docking capacity to hold the unit in question (either).

Refuel/UNREP/Replenish (if possible): The unit will head over to a tanker, UNREP ship, or ammo truck and refuel/resupply. The tanker/replenish unit can be selected either automatically, manually, or from a certain mission. This is essential if one needs to ration fuel or prioritize one specific mission over another (ie, the heavy bomber that needs the fuel for a deep strike over short-range fighters who'd only gain a bit of endurance). Submarines are also able to resupply and replenish assuming the supplier has the appropriate munitions and fuel types on board. Note that this can take a significant amount of time at sea due to the complexities of loading items onto a rolling submarine.



Hold Position - Selected Unit: For land units, this stops the selected unit(s). This is useful if they've moved to the wrong place.

Hold Position - All Units: This stops all land units.

Quick Turnaround (Airborne Aircraft): This allows the quick turnaround settings of an airborne aircraft to be reconfigured.

Summon to re-establish comms: If “Realistic Submarine Communications” (see 5.4.9, Scenario Features and Settings) is checked and a submarine is currently too deep to have normal communications, selecting this will cause it to, if possible return to a depth where it can communicate constantly.

4.1.4 Group Operations:

Group Selected Units: The multiple units selected will be unified into one group.

Detach Selected Units from Group: If one or more units in a group in Unit View is selected and used, it will be removed from the group.

Formation Editor: This opens the formation editor to change the placement of units inside the group. See “Formation Editor” under “Buttons and Windows” for more information

Unassign Unit(s): This unassigns the selected unit(s) from the mission in question.

Assign To Mission: The assigns the selected unit(s) to an existing mission.

Doctrine/ROE/EMCON: This brings up the doctrine window to allow unit level doctrine, ROE, EMCON settings and WRA to be set. (See Doctrine, under Buttons and Windows)

Range+Bearing Tool: This opens up the range/bearing tool. Clicking on a point starts the process, with a line being created. The line can be moved across the map, and in doing so, the distance and bearing from the original point appear next to the cursor. Clicking again creates a new “start point” at the cursor’s current location, while pressing ESC, CTRL+D or double clicking exits the tool.

Unload Cargo: This makes the unit, if able, unload any cargo that it’s carrying. Ships need to be right next to the shore to unload their cargo, while helicopters will need to drop down and land.

Select Unit(s) to pickup: This enables a suitable unit to pick up units and carry them as cargo. When selecting this, the cursor turns into a crosshairs. Click on the unit you want picked up.

4.1.5 Scenario Editor:

Edit Unit Properties: This allows unit properties-magazines, stores, and aircraft- to be changed. The time underway can also be changed, to simulate food and other non-fuel resources being consumed aboard a ship/submarine.

Set Unit Proficiency: Allows the proficiency of the unit to be changed. For more information about unit proficiency, see 5.4.4, Unit Proficiency.

Chance Of Appearance: This sets the probability that the unit will appear when the scenario is next loaded. Thus, if it’s set to 10, there’s a 1-in-10 chance that it will appear on load and a 9-in-10 chance that it will not.

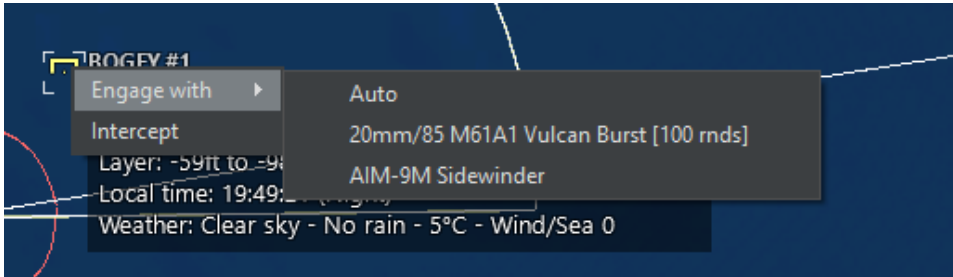
Copy Unit ID to Clipboard: Copies the unit ID to the clipboard where it can be used in Lua scripts.

4.2 Control and Shift Right Click on Map Dialog:

You can also Ctrl button + right mouse click on an empty spot on the map for some other functions.

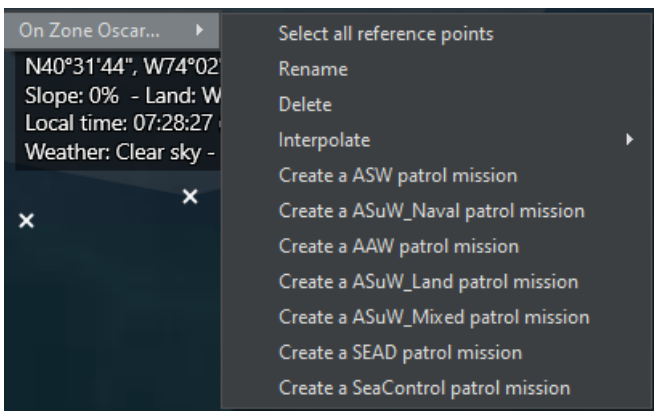
- **Add Unit** appears in ScenEdit mode and calls on the add unit dialog to drop a unit where you clicked.
- **Add Reference Point** allows you to drop one reference points where you clicked.
- **Define Area:** Allows you to, via click-and-drag, make a “box” of reference points. Areas defined can be rectangular or circular. To define an area, left-click and hold after selecting. If a unit is selected, the distance from the selection will appear in the databox. Once the area is large enough, release the mouse button and the box/disc of reference points will be formed.

If a potential target is right clicked while holding the shift key, a new dialog appears:



This allows you to engage a target with the selected unit using specific weapons or to simply order an intercept. If an intercept is ordered, the selected unit automatically selects an appropriate throttle settings and moves to identify the target using the best path possible.

If a unit that is capable of carrying and unloading cargo is selected and the shift + right click option is done on a unit that can be carried, a new menu will appear allowing the carrying unit to pickup that unit. If a base is shift-right clicked, it can be set as the new home base. If a unit that has a base is shift-right clicked, then the option to make that unit RTB is presented. If a unit is selected and a pre-existing zone is shift-right clicked, a new dialog appears:



This dialog menu allows you to create quick missions by selecting the appropriate option. Once you pick an option, the mission editor dialog will appear and allow you to dial in the appropriate details for that mission you just created. This can be done with individual units or groups, as appropriate.

4.3 Units, Groups and Weapons Symbols

Units can be utilized in the game as individual units or groups. Groups are useful to create as you can give many units common orders and you can organize ships into formations for a better common defense.

To group units simply drag-select the units you would like to group together and press the G key. You will now notice they have merged into one group symbol, which you can rename to whatever you would like.

If you'd like to remove members or dissolve the group:

- In game mode, you can detach individuals by selecting them in unit view (see below) and pressing the D key.

Once you get down to one unit the group will dissolve.

- In editor mode you can select the group and press the delete key once which will dissolve the group.

Units and Groups are displayed on the game map as color-coded symbols. The colors note postures (player side, friendly, unfriendly and hostile) and the symbols themselves let you know what kind of units they are.

Symbols on the map are also shown with a small letter F indicating that the contact is friendly to the player and a small letter A meaning that that contact is allied with the player.

All friendly units that have friendly postures set from both sides have an A displayed on the bottom left-hand side of the icon.

Unit Aimpoint Counts

The small white numbers next to units are a mount and sensor aimpoints within the unit. This allows players to know how many components (ex. Each tank of a 4-point tank platoon) are within a unit so they can allocate the correct number of weapons.

Reference Points: Reference points are markings that can be used as area boundaries, waypoints or simple markers. There are several types with different properties.

Unselected Reference Point: The reference point is on the map and is functional but is not currently selected.

Selected Reference Point: The reference point is currently selected. Its name appears on the display. When creating a new mission that requires them, all selected reference points will make up the default mission course.

Locked Reference Point: The reference point can be selected or deselected, but cannot be moved by the player.

4.4 Group Mode and Unit View Mode

You can toggle between unit view and group view by pressing the **9 key** on the keypad of your keyboard. To check which mode you are in please check the value on the lower left hand side of your display, where it will say "**SWITCHED TO [UNIT/GROUP] VIEW .**"

Group view is default and displays the group symbol and group datablock for units that have been grouped together. You can see individual units in a group when it is selected but cannot select and give individual orders to those units.

Unit view displays every unit on the map. You can select and give individual units orders in this window.

4.5 Right Side Information Panel

The dialogs on the right side of the display are the unit/group information dialogs. These are utilized by left clicking a unit/contact in the display which changes the different dialogs displaying information about the selected units/contacts. What and how much is displayed depends on what the player has access too. If the unit is on the player side, friendly or the game is in scenario editor mode the player will have access to all information. If in game mode and the player is neutral, unfriendly or hostile the player will only have access to what their contact information filters too.

The right side information panel can be extended or retracted via the button on the upper left side of it.

4.5.1 Unit Status Dialog

Name: The unit or group name will be displayed at the top. If an image for the unit is available, it will be remotely downloaded and displayed.

Proficiency: The unit's proficiency(see 5.4.4, Add/Edit Sides) will be displayed.

Type: Clicking on this opens the database viewer with the selected unit displayed. If a group is selected, clicking will produce a list of the group's composition. Clicking a unit name on the list opens the database viewer with that unit displayed. Multiple database viewer windows can be opened at once should it be desired, such as if the player wishes to see the capabilities of all the escort ships surrounding a carrier.

Group Composition: If the selected unit is a group you can click the down arrow button to display the member classes of the group. Clicking on individual entries will shift to unit view and select the unit in question.

Unit Message Log: Clicking this button brings up a message log. This functions similarly to the side-wide message log, except it only deals with messages concerning that unit. So it will only show contacts the unit personally detected, weapons calculations dealing with what the unit personally fired, damage reports dealing with what the unit personally took, etc...

Contact Report: Clicking this button opens up the contact report dialog that presents a list of detected emissions (sensor), a list of possible unit matches, and if applicable, units hosted. This is helpful in identifying the unit. Clicking on the various entries in the possible matches list will launch the platform display giving players full information on the unit.

The "Emissions" tab will display detected emissions and possible matches based on them (IE, an AN/AWG-9 radar will almost certainly be an F-14 Tomcat of some sort). The "Spotted Hosted Units" tab will display whatever units parked at the contact that have been seen by friendly optical sensors. See 9.3.1, "Building and Destroying Air Bases" for more details on hosted units.

Loadout: If applicable, this displays the loadout of the selected air unit.

WRA: The contact's Weapon Release Authorization type (see 3.3.15, Weapon Release Authorization) is displayed. An unidentified contact is displayed as the appropriate surface/air "unknown type", while a positively identified one is displayed as what fits its setting—an F-16 is a "4th gen fighter", an E-3 is an "Airborne Early Warning and Control", a giant oil tanker is a "Ship-Merchant/Civilian, 95000+ tons", etc...

Side: The unit or group side is displayed.

Course: Indicates the current course for a mobile unit or facing direction for a stationary unit or fixed facility.

Speed: Indicates the unit's current speed.

Group lead/lead can slow down (group reform) checkbox: When selected, allows the formation leader to slow down when it is necessary for group members to move to regain their position in a formation. This prevents group leaders from breaking away from the main group and is selected by default.

Damage: Shows the Total amount of damage the unit has at this moment.

Damage Ctrl Button: Opens the Damage Status window. See "Systems and Damage" under "Buttons and Windows" for more information.

Fire and Flood Meters: Display the fire and flood damage as it occurs. The more the flood or fire bar is filled, the more damage over time it does to the affected unit, and when the bar fills all the way, the unit is automatically killed. The ability to control fire and flooding depends on a variety of factors, including side proficiency. Compare the crews who saved the Samuel B. Roberts from the mine hit when every simulation had the ship breaking in half to the fates of the Taiho and Shinano carriers, doomed from seemingly survivable hits.

Systems: Shows the damage to various systems (weapons, sensors, radios, engines, etc...) on the unit. As anyone who knows the meaning of the term "mission kill" can say, a unit can have its hull intact and yet be unable to meaningfully fight.

Assigned Base: Shows the assigned home base of the unit, if any.

Status: Displays the status of the unit, such as "unassigned" (doing nothing), "engaged offensive/defensive" (engaging in battle), or "on plotted course" (moving)

Mission: Shows the mission the unit is currently assigned to, with a link to the appropriate entry in the mission editor (see 7.1, Mission Editor).

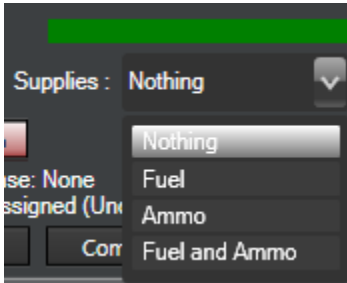
G-Tolerance: For aircraft in high-g situations such as turning dogfights or evading missiles, this shows the amount of strain on the pilot. When full, the aircraft is no longer able to make as tight turns as it could previously.

Last Detections: Displays the unit that last detected the contact in question and the sensor on said unit that did so as well. Clicking on the unit name automatically centers the camera on the detecting unit, while clicking

on the sensor name brings up the database viewer of the detecting unit with the sensor in question highlighted in green. The “Last Detections” menu can be folded up or down by clicking on its bar.

Magazines: Pressing the magazine button launches the magazine dialog which displays individual magazines, weapons records and number of weapons carried. For more information, see “Magazines” under “Buttons and Windows”

Supplies: This menu allows you to define the type of supplies that a unit can provide to a given group or to the rest of a side/allies. Selecting options from this menu can limit a provider to only providing for its group rather than all possible receivers.



Boats Button: Displays the number of available boats and total boats in the selected unit. Pressing the button launches the Docking Ops Dialog. For more information, see “Boat Operations”, under “Messages and Windows”.

Edit Hosted Boats: Only available in ScenEdit for units capable of holding and deploying boats, clicking it “Edit Docked Boats” window. Once there, boats can be added or removed as needed.

Aircraft Button: Shows the number of available aircraft and total aircraft in the selected unit or selected group. Clicking the button launches the Air Ops dialog which is used to manage hosted aircraft for the base in question. For more information see “Air Operations” under “Buttons and Windows”

Edit Hosted AC: Only available in ScenEdit to units capable of deploying aircraft, clicking this brings up the Edit Aircraft window. Once there, aircraft can be added or removed as need be.

4.5.2 Sensors Button

Pushing this button launches the sensor dialog that displays a list of all sensors on a unit. See “Sensors” under “Buttons and Windows” for more information.

4.5.3 Weapon Buttons

Pressing this button launches the Weapons Dialog which displays information on all weapon mounts on a platform. For more information, see “Mounts and Weapons” under “Buttons and Windows”.

4.5.4 Unit Fuel

A green bar that shows the amount of fuel remaining. All units consume fuel except facilities and land units. Units with nuclear power plants will not run out of propulsion fuel. Fuel consumption rate is a function velocity for all units and is not a linear relationship. A/C have fuel consumption modifiers for altitude and throttle setting including afterburner

Total units remaining and time until expended are displayed below. These will change based on consumption.

4.5.5 Unit Alt/Speed

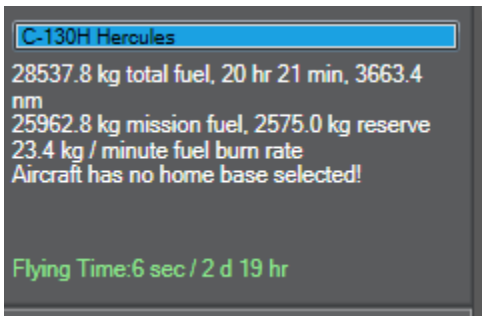
When expanded, this allows for the unit's speed and (if applicable) altitude to be quickly changed.

4.5.6 Unit Fuel

When expanded, this displays the unit's fuel. For surface ships, it involves the remaining amount and time to run out at current consumption. For aircraft, it is more detailed and involves mission fuel, reserves, burn rate, distance to its home base, how far away (in terms of time and quantity) it is to bingo fuel, and how long it's been in the air (important for quick-turn loadouts and seeing how much its crew can take before they get tired and have to return to base to sleep).

Pressing the "jettison" button, if available, will release whatever drop tanks the aircraft has, increasing its maneuverability at the obvious expense of range. Use this with care when far away from a base or tanker.

For air units, there is an additional item for flying time:



All aircraft have a limited amount of time they can be airborne before they are forced to immediately return to base. This is determined by the experience of the unit and the size of the crew. This means that it is not possible to keep an aircraft in the air indefinitely regardless of how it is refueled.

4.5.7 Unit EMCON

When expanded, this contains shortcuts to the EMCON and sensor windows, as well as buttons that can set radars, sonars, and/or jammers to active or passive.

4.5.8 Doctrine

When expanded, this contains a link to the doctrine window, as well as the ability to change any particular ROE for the unit right there.

4.5.9 Doctrines, Postures, Weapons Release Authority, and Rules of Engagement

Doctrine & RoE: Shows the unit/groups Doctrine and Rules of Engagement settings.

- **Change Button:** Click this to open the Doctrine & RoE/ EMCON dialog, described below.
- **Doctrine & RoE Dialog:** This dialog lets you set Rules of Engagement (RoE) settings for units and groups. Doctrine and Rules of Engagement govern platform behavior, or posture, towards other platforms in the

scenario. The Scenario Designer sets the initial Doctrines and RoEs and may enable the player to edit one or all of these. For more information, see “Doctrine” under “Buttons and Windows.”

5.0 ScenEdit

The scenario editing mode in COMMAND is accessible by selecting either “Create New Scenario” or “Edit Scenario” from the start menu, or the “create new blank scenario” option from the file menu. Existing files can be accessed via the “Edit Scenario” choice on the main menu.

5.1 Getting Started

The scenario editor is the most versatile part of COMMAND. It can be used for making scenarios, with V 2.0 boasting a rewritten map engine and other “under the hood” performance improvements that make building and running monstrously large battles easier than ever. It can also be used for “editor experiments” where certain situations can be quickly set up for the sake of entertainment or curiosity.

Certain features can only be used in it, or in “ScenEdit” mode. This section will explore those features.

5.2 Scenario Intro Walkthrough

To make a scenario.

Basic:

- Click “**Create New Scenario**” on the start menu. This will bring you to a blank globe screen.
- Go to “**Add/Edit Sides**” on the editor menu. Then click “**Add**” and name the first side.
- Go to “**Unit Actions**” and then click “**Add Unit**” on the editor menu, or press the **INSERT** key.
- Select the type of unit(s) to add (see 5.7.1, Add A Unit). Make sure to add airbases for air units.
- Return to Add/Edit Sides and add additional sides, particularly the enemy side. If you do not wish for them to be playable, make sure to select the “**Side Is Computer Only**” button.
- Add missions via the “**Missions/Ref Points**” menu (see 7.2, Missions). For non-playable sides, this is necessary-otherwise they will just sit there and do nothing. For playable sides, it can still be useful.
- Once the missions are added, saving the scenario is highly encouraged.
- A test run to see if the missions work is also strongly encouraged, reloading to the original file.
- If everything works as intended, proceed to add scoring events (see 5.5.5, Constructing an Event)
- Save the scenario, and then test-run it to ensure the scoring events work.
- Add briefings and scenario descriptions.

5.2.1 Full Scenario Walkthrough

- Create a scenario idea.
- Perform research if necessary. The need for historical accuracy varies greatly on both the designer's preferences and type of scenario. Obviously, a detailed re-enactment of a real battle needs to be highly researched. A looser alternate history can be but does not have to.
- Go to the editor and load an appropriate database. Depending on the time period, it's either the Cold War Database or DB3K. **CWDB is WWII to 1980, DB3K is anything beyond that.**
- Create the sides of the scenario and set their postures and conditions to an appropriate setting. Not only should hostile sides be marked as such, but *neutral traffic should be set to "Blind" to reduce CPU stress.* (See 5.4.4, Add/Edit Sides)
- Decide which side the player will be. Set non-player sides to "Computer-Only"
- Add the units and, if necessary, the missions for the player's side.
- Add the units and missions for the opponent's side. This is necessary for a computer opponent.
- Add the necessary events and scoring. (See 5.5, events)
- Playtest the scenario.
- Complete the final checklist. (see immediately below)
- Submit the scenario.

5.2.2 Final Checklist for Polishing

- Scenario saved as a scenario file (scenarioname.scen) not a save game (savename.sav).
- Scenario Times, duration, complexity, difficult and location/settings are set.
- If a playable side has tons of sensors the merge range-symbols in map settings is turned on to reduce clutter.
- Scenario description and the scenario briefings for playable sides are written.
- Sides are added, postures, ROE and awareness are set. If side is not playable the computer only check box is checked.
- If a side is to be playable by both the AI and a human player than all missions created for the AI have the Scrub mission if human side is human checkbox checked in the mission editor.
- If a side is included just for environment (fishing boats, merchants) you've set their awareness level to blind to minimize unnecessary sensor calculations.

- Units are named, and they are assigned to missions.
- The scenario features and settings are set to what the designer intended.
- Air bases and ships that host aircraft have populated magazines. Alternatively, the “unlimited ordnance at bases” option is set to default to on.
- Aircraft are armed or set to reserve, so players know to add munitions. Aircraft that are not to be used are set to maintenance.
- Boats, landing craft and UUV are added where appropriate.
- Groups are named, and their formations are set in the formation editor.
- All reference points are deselected you don’t want selected and they are locked if you don’t want players moving them.
- No-navigation zones are added and locked where necessary.
- Missions and exclusion zones do not share reference points. Likewise, missions and no-navigate zones do not as well.
- All events created in the event editor have triggers and actions.
- If added, special actions work properly and effectively.
- If you are using points in the scenarios you have correctly configured the scoring dialog.
- If the game is in unit view switch it back to group view by pressing the Page up button.
- Layers (Relief, latitude and longitude) are activated in map settings if you would like to use them.
- God’s eye view is turned off if used.
- If you have a message displayed you’ve cleared them by select message log and clear for the game drop down menu.
- The losses and expenditures report is clear. Any erroneous entries can be cleared by clicking the Reset All button.
- The scenario elements work properly and there are no in-game problems (ie, a missile strike that never activates not because of a bug or scenario editor flaw, but simply because its target is never spotted in time)
- The scenario feels fun to play. A scenario that is too hard (the victory condition can only sometimes barely be met) or too easy (the player’s side can effortlessly win with standoff weapons) may need reworking even if everything’s gone right technically.

5.2.3 Detailed Single Scenario Walkthrough

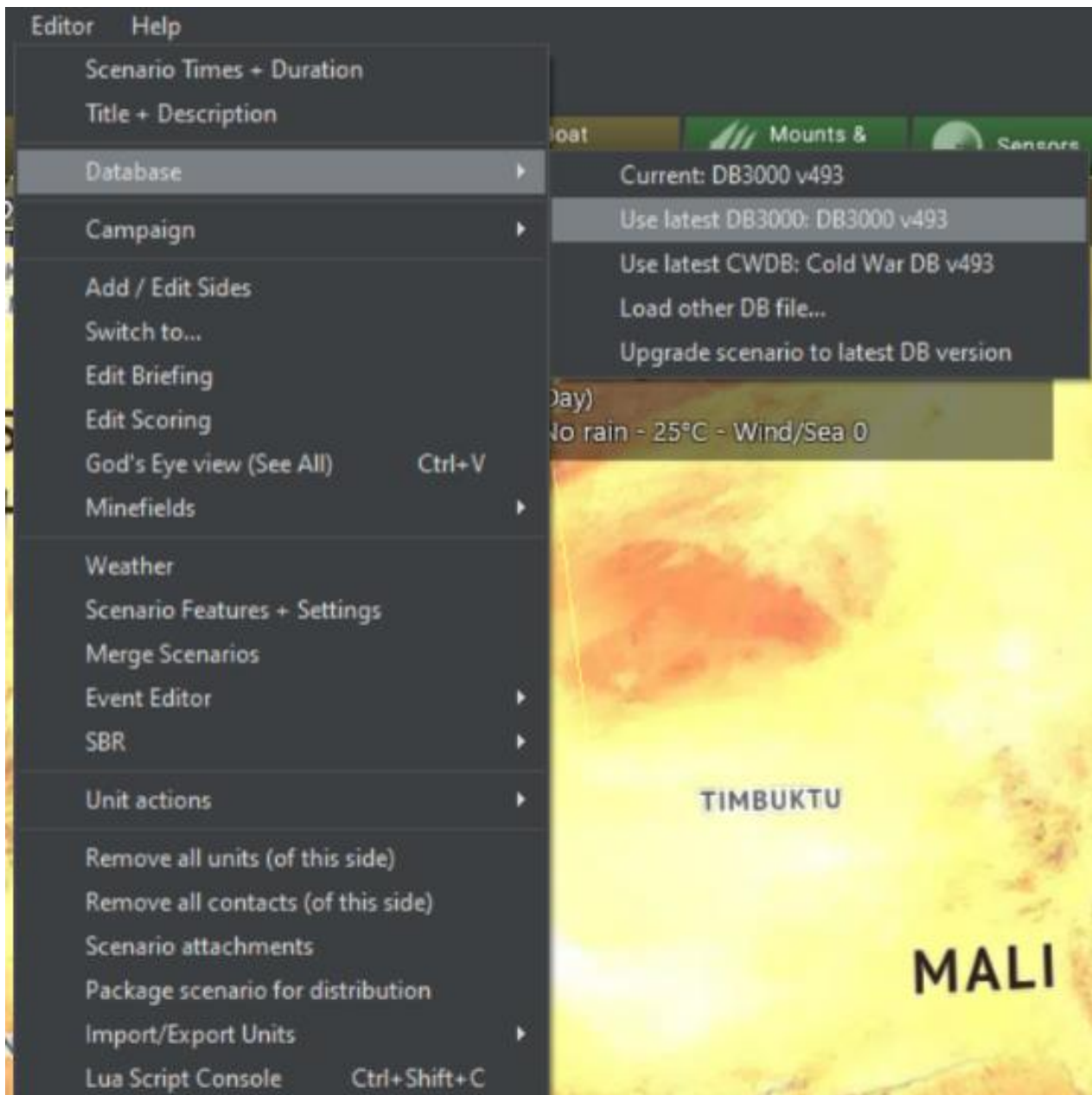
This section goes into detail on the exact steps needed to make a very simple scenario.

-Click on “Create New Scenario” on the title screen or “Create New Blank Scenario” on the file menu. A blank world map should appear.

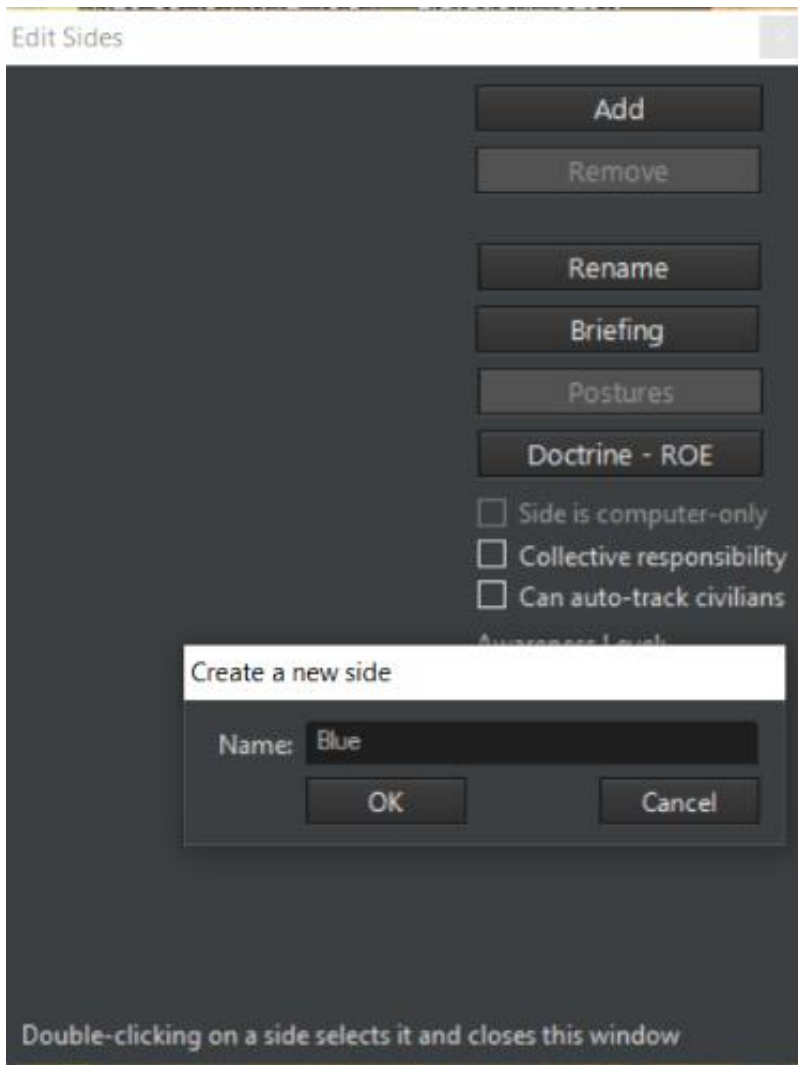
-Move and zoom the map to the area of the world you want the scenario to take place in. While COMMAND allows for intercontinental travel, a beginner should start small. This step doesn’t *have* to be taken now, but it doesn’t hurt. Let’s zoom in on Mauritania.



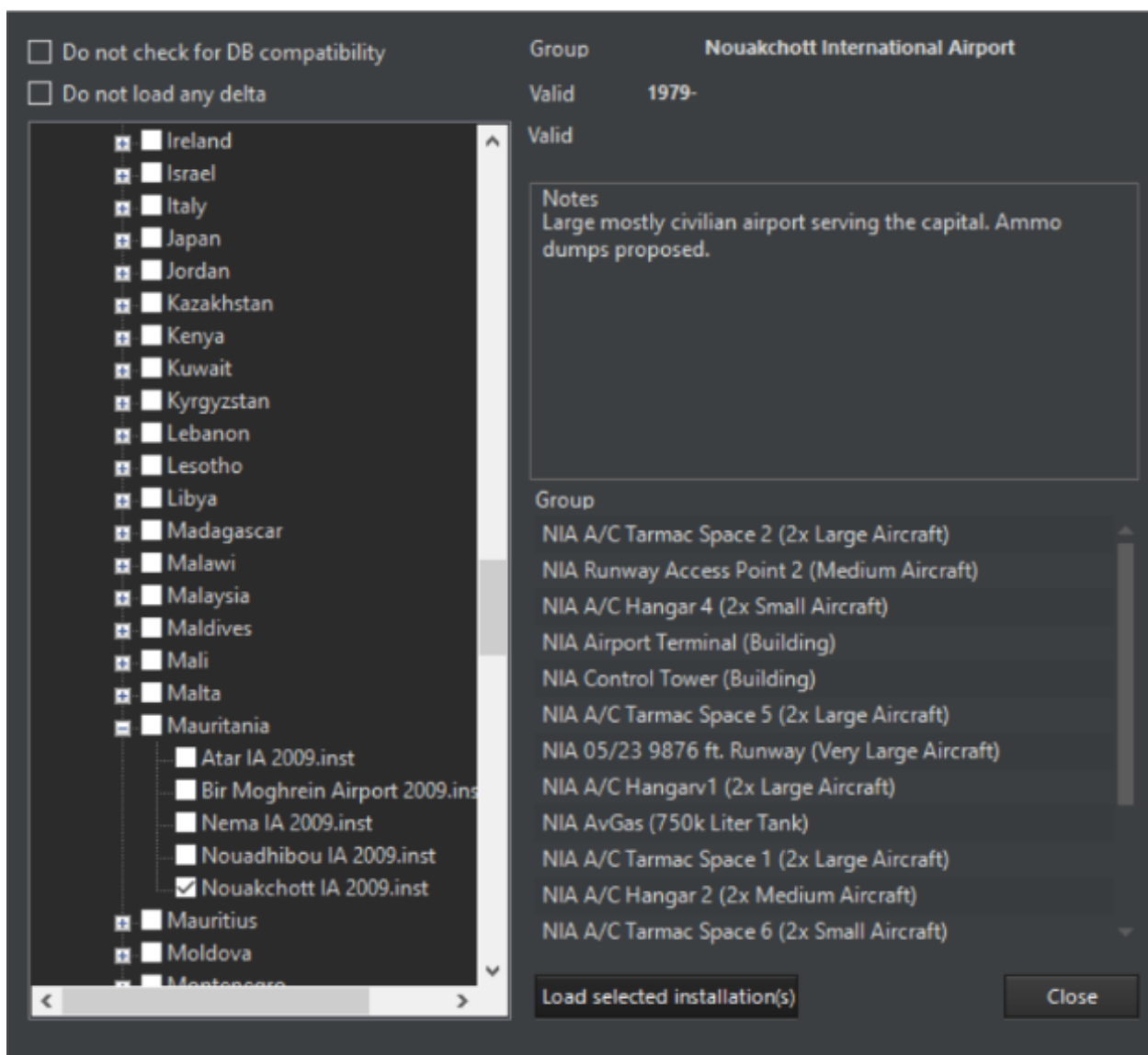
-Go to the Editor drop down menu. The first thing you should do is check the database. Since changing DBs wipes the rest of the scenario, the database should be decided on before any more work is done. There are two main databases, the DB3000 (1980+) and the CWDB (1946-1979). The DB3000 won’t have World War II surplus fighters like P-38s, and the CWDB won’t have advanced modern aircraft like F-22s. Here we want the DB3000, so select “**use latest DB3000.**”



-Now add a side. This will be the player's side. We'll just call it "Blue" for now. Go to Add/Edit Sides on the editor drop down menu, click add, type "Blue", and click "OK."



It's time to add an airbase. Fortunately, COMMAND has many prebuilt ones. Go again to the **editor drop-down menu**, then go to "**import-export units**" and "**load units**". Go to "**main, import/export**", then scroll down to "**Mauritania**" and expand. Check the box that says "**Nouakchott IA**", and then press "**Load Selected Installation(s)**". Then close the window after it says "**completed!**"



A small airfield shall now appear on the Mauritanian west coast. Click on it.



Go to either “**Edit Aircraft**” in the Editor drop-down menu or press the “**Edit hosted aircraft**” button on the right bar with the airfield selected. The edit aircraft window will appear.

Into “Class”, type “F-4S”.

Two types of (functionally identical) units will appear. Select the lower (Navy) one, type “**Phantom**” for callsign, and enter **5** into “how many”. Then press “**Add Selected**”. Five F-4S Phantom IIs should appear in the “current” tab on the left. Close the window.

Current

Aircraft to

Filter by...

Class: F-4S

Country: None

Hypothetical: Both real-life and hypothetical

SubType: None

Aircraft	Country	From	Until	Hypo	Type	Deprecated
F-4S Phantom II -- United States (Mari...	United States	1978	1986	☐	2002	☐
F-4S Phantom II -- United States (Navy...	United States	1978	1986	☐	2002	☐

Quantity: 0

Apply

Callsign: Phantom

How many: 5

Add Selected

Now zoom close to the airbase. Go to either Editor Drop Down/Unit Actions/Add Unit or just press insert. The mouse cursor should change. Click on an area of land near the airbase.

Select **Facility**, then in the “class” box, type in **Bar Lock**. Several types of Bar Lock radar will appear. Select one and press **OK.** A Bar Lock 3D air search radar station will appear at the spot. Click on the Bar Lock to select it. Note that in the Add new unit dialog, you have the ability to filter the desired units by class, subtype, country and whether the unit is real or hypothetical.

Add new unit
✕

Type: Facility
Side: RED Edit sides...
Name: Radar (Bar Lock A [P-37])

Filter by...

Class: Bar Lock
Country: None

SubType: None
Hypothetical: Show all platforms, both real-life and hypothetical

Custom GUID:

Platform	Country	From	Until	Hypo
Radar (Bar Lock A [P-37]) -- Angola (Air Force)	Angola	0	0	☐
Radar (Bar Lock A [P-37]) -- China (Air Force)	China	0	0	☐
Radar (Bar Lock A [P-37]) -- Czechoslovakia [-1992] (Air Force), -1992, CGI	Czechoslovakia [-1992]	0	1992	☐
Radar (Bar Lock A [P-37]) -- Germany [DDR, -1990] (Air Force), -1990, CGI	Germany [DDR, -1990]	0	1990	☐
Radar (Bar Lock A [P-37]) -- Germany [FRG/Reunified] (Air Force), 1990-1997, CGI	Germany [FRG/Reunified]	1990	1997	☐
Radar (Bar Lock A [P-37]) -- Iraq (Air Force)	Iraq	0	0	☐
Radar (Bar Lock A [P-37]) -- Libya (Air Force)	Libya	0	0	☐
Radar (Bar Lock A [P-37]) -- Lithuania [1992-] (Air Force), 1997, 4x	Lithuania [1992-]	1997	0	☐
Radar (Bar Lock A [P-37]) -- Myanmar (Air Force)	Myanmar	0	0	☐
Radar (Bar Lock A [P-37]) -- Soviet Union [-1991] (Air Defence Troops [PVO]), -1991, CGI	Soviet Union [-1991]	0	1991	☐
Radar (Bar Lock A [P-37]) -- Syria (Air Force)	Syria	0	0	☐
Radar (Bar Lock A [P-37]) -- Yemen (Air Force)	Yemen	0	0	☐

OK
Cancel

On the right bar, press “**Sensors**”. Uncheck the “**Unit Obeys EMCON**” box and click the “**active**” box on the “Bar Lock A” option. Close the window. The radar is now emitting.

Sensors for: Radar (Bar Lock A [P-37])
✕

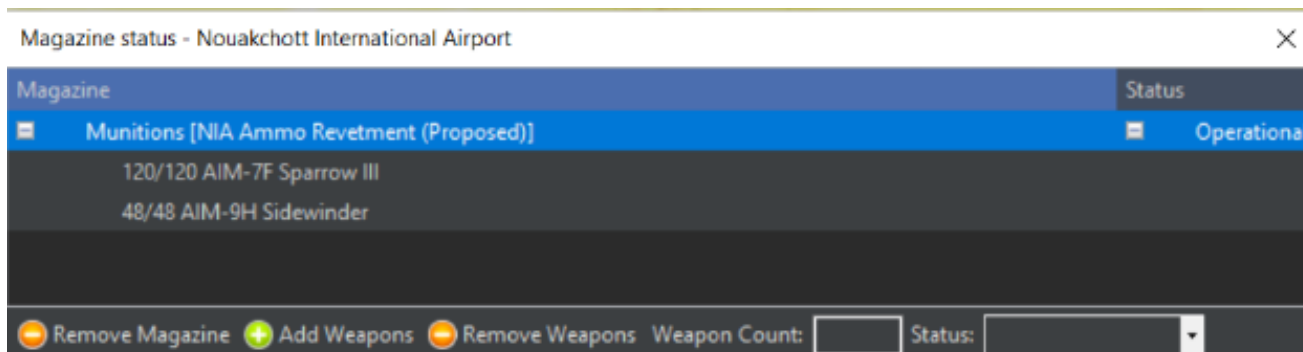
☐ Unit obeys EMCON (disables manual sensor control)

Quick selection: ☒ Radars ☐ Sonars ☐ Offensive ECM

Sensor	Sensor Type	Active	Status
Bar Lock A [P-35]	Radar, Air Search, 3D Long-R...	☒	Operational

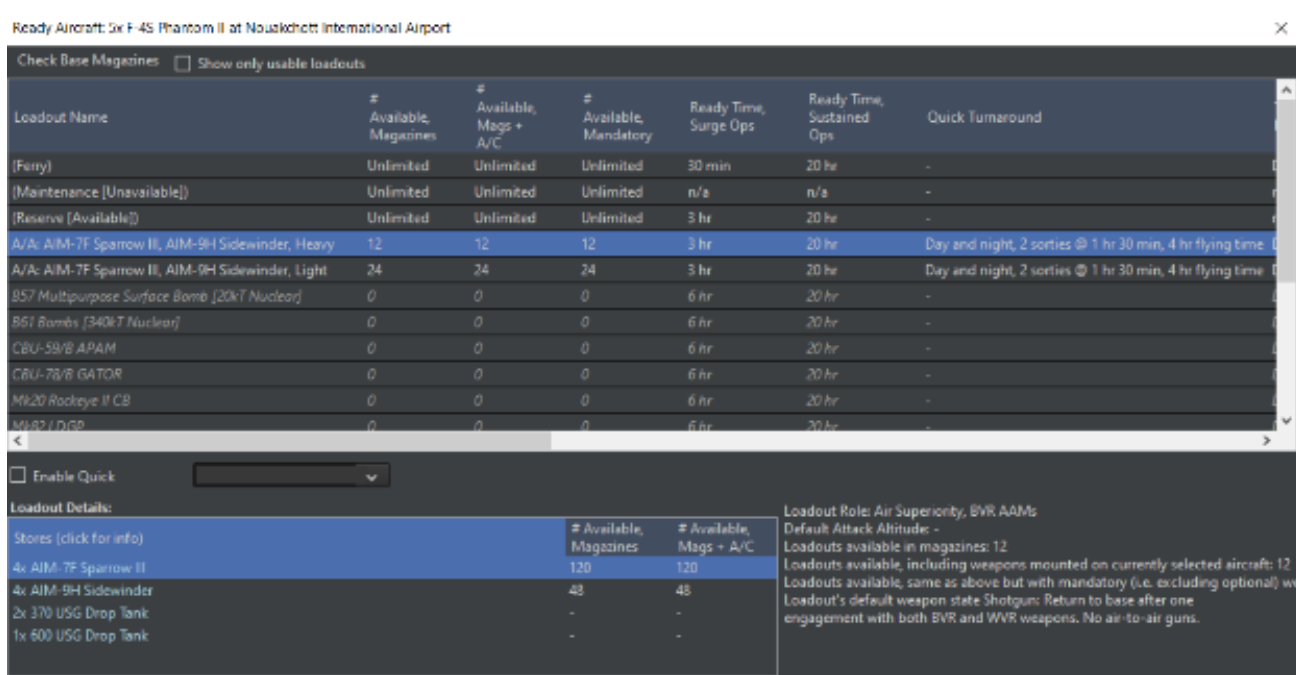
+ Add Sensor
➔ Remove Sensor

Now we need to arm the fighters. Reselect the airbase. Click on “**magazines**” on the right bar, then “**add weapons**”. Add the 120 Sparrows and 48 Sidewinders (or any number, but those choices are the most convenient).



Close the windows and click on **“aircraft”** on the right bar. The five Phantoms are all red with **“no loadout”**, meaning they cannot function. Select all of them, then press **ready/arm**. You’ll notice in the air ops window that the air to air loadouts are functional, but the rest aren’t. This is because only the air to air missiles are available in the magazines.

Select **“A/A: AIM-7, Heavy”**, and press the red **“ready immediately (ScenEdit) button”**. While in normal mode they’d need time to prepare, here they’re ready instantly. Now they appear as green, meaning they’re ready to go.

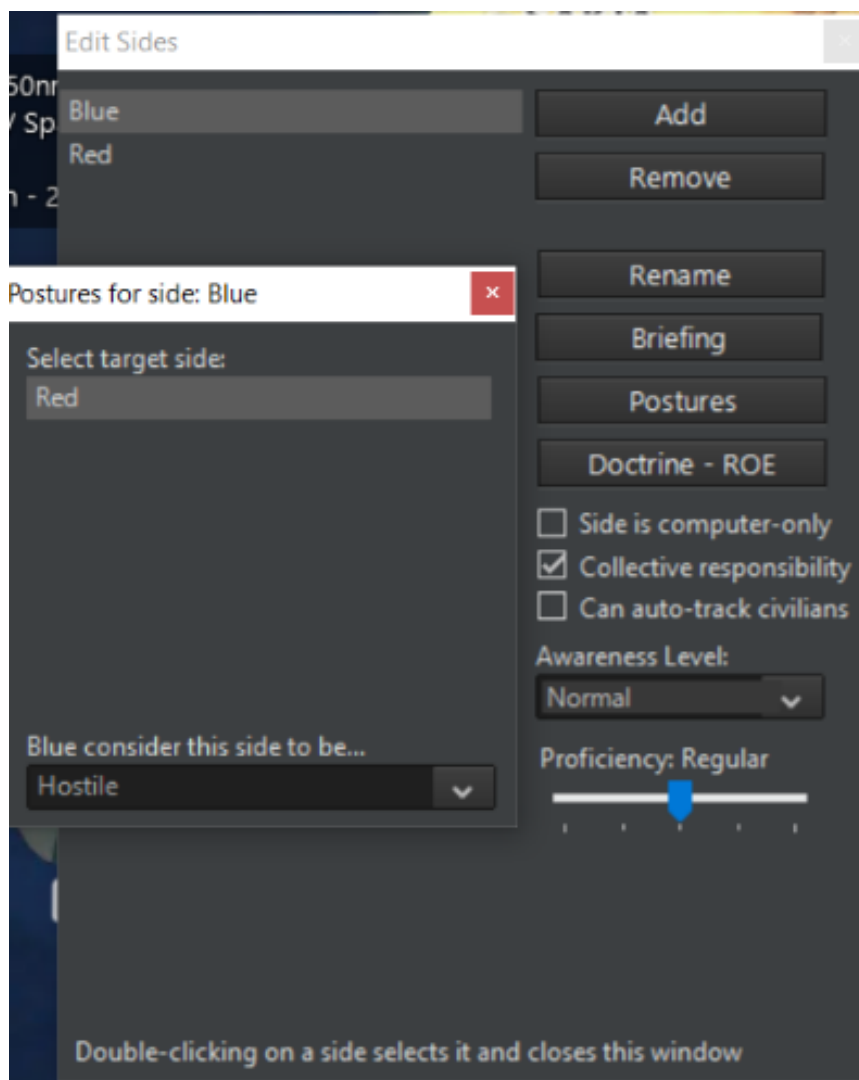


All right, one side is up. It has a radar station for spotting and aircraft capable of intercepting. Now for the other.

Go to **Add/Edit Sides** again, then add a side called **“Red”**. Before closing the window, make sure to check the **“side is computer-only”** box. Now go to **“Switch To”** and select Red.

The map is now blank again. Go to add unit/insert, click on a spot near the northern Mauritania/Western Sahara border, and type in “**Single-Unit Airfield**”. Select one with a long runway and press OK. It should appear.

Now press the play button for a few seconds and then pause the simulation again. Two green symbols-the Nouackchott airbase and Bar Lock, should appear. Green means neutral, and we want these to be hostile. Go back to add/edit sides. Click on Blue, go to Postures, and then select Hostile. Do the same for Red. They should now appear as red on the map.



Select the airfield (making sure you’re controlling Red) and then click “**edit hosted aircraft**” on the right bar. Type in C-130A and select one of the two C-130As (NOT the AC-130A on the top). Write “**Target**” under callsign and add two.

Go to “**planes**” and ready/arm again. Select **Ferry** and **ready immediately**. There’s no need for filling the magazines in this case.

They need missions to function. Using **CRTL-right click** or **CRTL-Ins**, put two reference points near, but not too near, the Blue airbase. Make sure they’re selected.



Now go to the missions/reference points drop down menu and add mission. Make it a support mission.

New Mission / Task Pool / Package

Category: Mission

Name: Target Track

Class: Support

Type:

Status: Active

Parent pool:

☒ Open Mission Editor window

Activation Time

Date: None

Time: None

Deactivation Time

Date: None

Time: None

☐ Unassign Units

☐ Order RTB

☐ Delete Mission

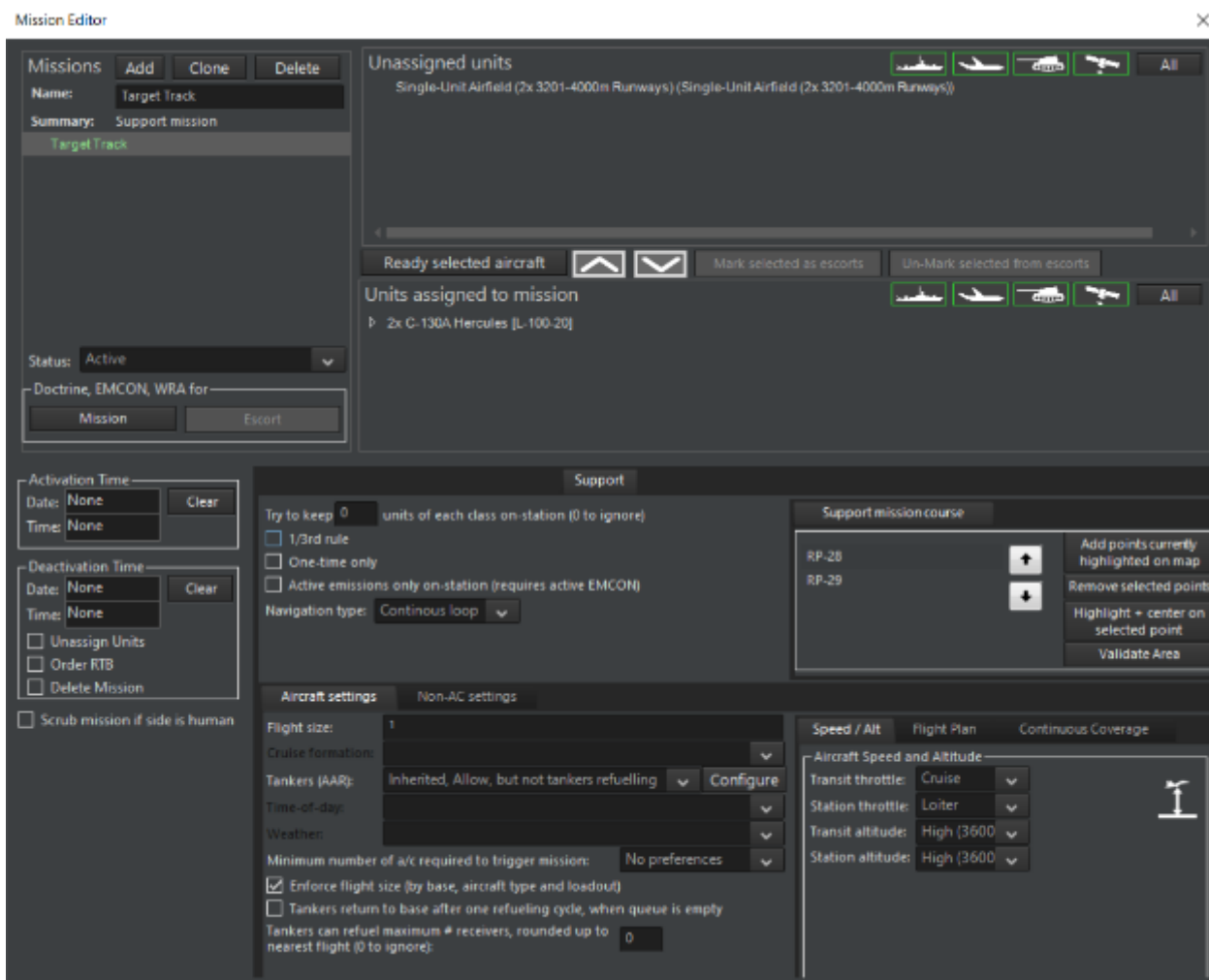
Clear

Clear

OK

Cancel

Select both C-130s and add them to the mission, then uncheck the 1/3rd rule.

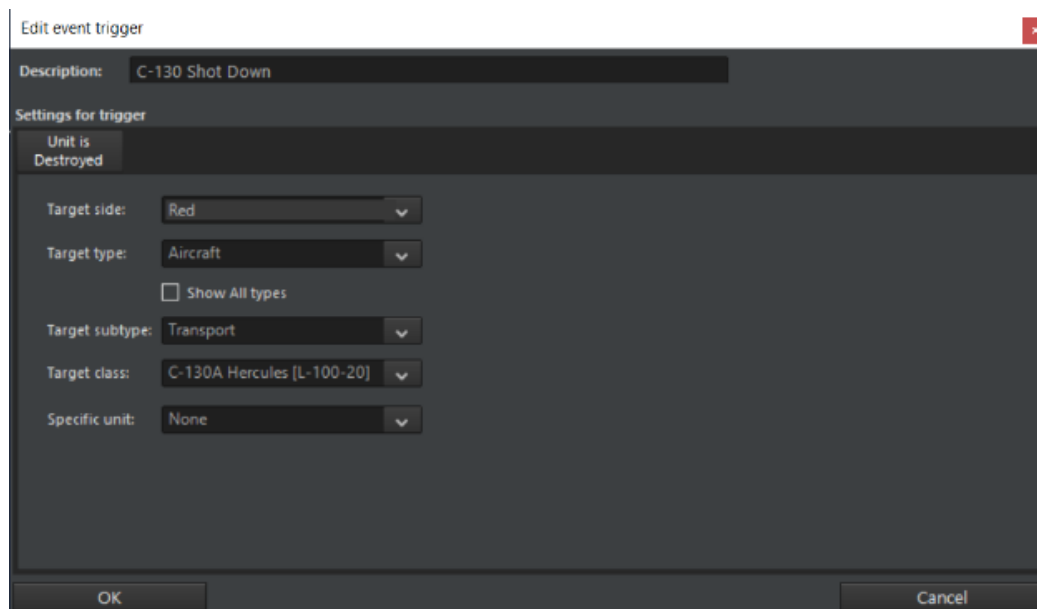


Now save the game. If loaded in normal play, the scenario now functions-the C-130s will take off on their own. But now it's time to make it meaningful.

Switch to Blue or exit and reopen the scenario (it will load as Blue by default, as Red is checked to computer-only). Go to Editor-Dropdown/Edit Scoring. Set the Triumph score to 150. Close the window.



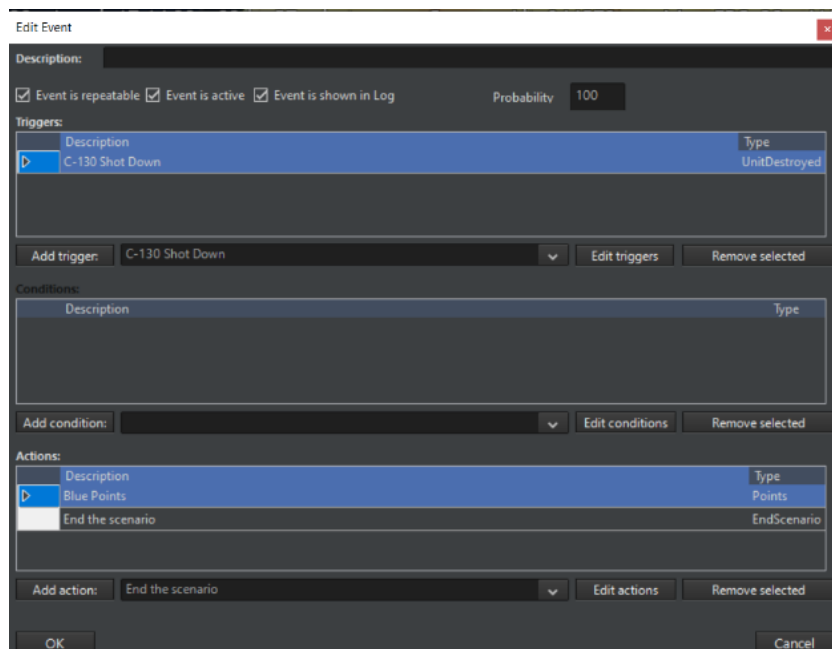
Now go to **Editor/Event Editor/Triggers**. Select a “**create new trigger, unit is destroyed.**” Type in C-130 Shot Down. Select Target Side Red, Target Type Aircraft, Target Subtype Transport, Target Class C-130A Hercules. Press OK.



Now Editor/Event Editor/Actions. You want two actions this time. Select “**Create New Action**” with “**Points**”, and then Side: **Blue, Point Change 500**, with **Blue Points** in the description. Press OK. Then select Create New Action with “**End Scenario**”. Press **OK** in the now limited action editor window.

Close the window, now go to Event Editor/Events.

Check the **Event is Repeatable** Tab, Type in **C-130 Hit for “description”**, then add **C-130 Shot Down** for trigger and “**Blue Points**” and “**End The Scenario**” for Action. Press OK. Save the scenario.



By this point, the internal mechanics of the scenario are complete. All that's left is touching it up.

At the very top of the editor drop down menu is “**scenario times-duration**”. Make sure the “**scenario current time**” and “**scenario start time**” are the same, set the “**complexity**” to 2 (for the sake of the exercise) and put “**Mauritania**” as the location. Then press **OK**.

Now go to editor menu-“**Title/description**”. This has probably already come up as part of the initial save of the scenario. In the description box, write “**Defend southern Mauritania from unknown Red contacts. This is an editor test.**”

Then press **OK**.

Now go to “**Edit Briefing**”, and type in the sentence. “**Investigate and Engage incoming air contacts.**” Press **OK**. Most scenarios need and have a longer briefing, but this works for now.

Save the scenario again and return to the title screen.

At the the title screen, go to “**start new game**”, and select the scenario in question. The scenario will appear as having Mauritania as the location, a higher complexity and difficulty score, and the scenario description of “Defend southern Mauritania from unknown Red contacts. This is an editor test.”

The only option is to play as Blue, and you will see the Blue “Investigate and Engage incoming air contacts” briefing. Enter the scenario and run it for a bit. Two air contacts should appear, detected by the Bar Lock.

Using either an intercept mission, patrol mission, or manual engagement, send the Phantoms up to engage the contacts. They will be shot down. Once one C-130 is downed, if all has gone right, a popup saying “The scenario has concluded. You will now be presented with the evaluation of your performance.” will appear. Press OK. Another window should appear, saying Triumph: Final Score 500” (or 1000 if both planes were quickly downed). You’ve won!

A complete scenario has now been made and played.

5.3 Lua scripting

Lua is a programming language that Command uses for advanced scripting. It is extremely versatile and can provide many bonuses. Even at their simplest, Lua scripts can add much versatility beyond the limits of what the internal event editor alone can provide.

The Basics:

Lua can be used to:

- Spawn, as opposed to teleport, units into the map during a scenario.
- Change the weather mid-scenario.
- Reassign units to different missions.
- Change side postures and doctrine.
- Add elements of chance so that the scenario is never the same way twice.
- Make scenarios detailed and varied.

Thus, Lua can be used for creating a basic scenario where the player starts with a random variety of units. It is here that a basic walkthrough will be conducted.

In either a subset of “**event actions**” or a “**lua script console**” in the event editor, there's a Lua menu. A basic set of “**recipes**” are there, see “**Command Lua API**” at <https://commandlua.github.io/> for explanations. One such recipe is **ScenEdit_AddUnit** (Table). When filled in, it looks like this:

```
ScenEdit_AddUnit({type='Sub', side='USSR', name='B-23', dbid='187',  
latitude='49.3812997111192', longitude='-36.8903957715054'})
```

This, if run with a side marked “USSR” in a DB3K database, will generate a Foxtrot-class submarine named B-23 in the middle of the North Atlantic.

NOTE: Lua can be very finicky, and a single character in the wrong spot can mess up a script. Also, be aware that values carry over across scenarios, so do not assume they will be the default ones.

That example is using Lua to duplicate something that can be accomplished with the traditional interface. The same task could be accomplished via finding the spot in the Atlantic and using the unit editor. A more complex task is to include one of three very different submarines.

```
a = math.random(1,3)  
if a == 1 then  
    ScenEdit_AddUnit({type='Sub', side='USSR',name='B-10', dbid='287',  
latitude='50.52351341355', longitude='-37.177899984719'})  
elseif a == 2 then  
    ScenEdit_AddUnit({type='Sub', side='USSR',name='B-15', dbid='278',  
latitude='50.52351341355', longitude='-37.177899984719'})  
elseif a == 3 then  
    ScenEdit_AddUnit({type='Sub', side='USSR',name='K-115', dbid='391',  
latitude='50.52351341355', longitude='-37.177899984719'})  
end
```

Here there is an even chance of one of the following three submarines being spawned.

B-10, a Whiskey (one of the earliest and least capable diesel subs)

B-15, a Juliett (a 1960s diesel submarine that still has long-range anti-ship missiles)

K-115, a Charlie II (a nuclear submarine with advanced Bazalt/Sandbox-style anti-ship missiles)

Tying this script to an event that is triggered after the scenario is loaded (see 5.5, Events) is the next step.

Once a player (with the “USSR” being a playable side) runs the scenario in non-ScenEdit mode long enough to trigger the event, they will receive one of the three submarines.

For a non-playable side, patrol orders need to be bundled-in for any mobile unit. For example:

```
ScenEdit_AddUnit({type='Sub', side='USSR',name='B-24', dbid='287',
latitude='51.52351341355', longitude='-38.177899984719'})
ScenEdit_AssignUnitToMission('B-24', 'eastpatrol')
```

Creating two different (and possibly) empty missions in the scenario editor, having the USSR be a *non-playable* side, and using a math.random script as follows:

```
a = math.random(1,2)
if a == 1 then
ScenEdit_AddUnit({type='Sub', side='USSR',name='B-24', dbid='287',
latitude='51.52351341355', longitude='-38.177899984719'})
ScenEdit_AssignUnitToMission('B-24', 'eastpatrol')
elseif a == 2 then
ScenEdit_AddUnit({type='Sub', side='USSR',name='B-24', dbid='287',
latitude='51.52351341355', longitude='-38.177899984719'})
ScenEdit_AssignUnitToMission('B-24', 'westpatrol')
end
```

..means that once the event is triggered, a Whiskey submarine will spawn in the same spot and conduct a mission in one of two areas. This sort of Lua randomization is useful for increasing the variability and replayability of scenarios.

Lua can do more than just set up scenarios. It can also make them more dynamic.

One Lua action is using the previously mentioned **ScenEdit_AssignUnitToMission** to either assign or change missions for units that already exist. Simple random selection can make the following units assigned to one of three possibilities.

```
a = math.random(1,10)
if a<5 then
ScenEdit_AssignUnitToMission('Ukrainian Flanker #1', 'NONE')
ScenEdit_AssignUnitToMission('Ukrainian Flanker #2', 'NONE')
ScenEdit_AssignUnitToMission('Ukrainian Flanker #3', 'NONE')
ScenEdit_AssignUnitToMission('Ukrainian Flanker #4', 'NONE')
ScenEdit_AssignUnitToMission('Ukrainian Flanker #5', 'NONE')
ScenEdit_AssignUnitToMission('Ukrainian Flanker #6', 'NONE')
elseif a>=5 and a<8 then
```

```

ScenEdit_AssignUnitToMission('Ukrainian Flanker #1', 'Close Patrol')
ScenEdit_AssignUnitToMission('Ukrainian Flanker #2', 'Close Patrol')
ScenEdit_AssignUnitToMission('Ukrainian Flanker #3', 'Close Patrol')
ScenEdit_AssignUnitToMission('Ukrainian Flanker #4', 'Close Patrol')
ScenEdit_AssignUnitToMission('Ukrainian Flanker #5', 'Close Patrol')
ScenEdit_AssignUnitToMission('Ukrainian Flanker #6', 'Close Patrol')
elseif a>=8 then
ScenEdit_AssignUnitToMission('Ukrainian Flanker #1', 'Aggressive Patrol')
ScenEdit_AssignUnitToMission('Ukrainian Flanker #2', 'Aggressive Patrol')
ScenEdit_AssignUnitToMission('Ukrainian Flanker #3', 'Aggressive Patrol')
ScenEdit_AssignUnitToMission('Ukrainian Flanker #4', 'Aggressive Patrol')
ScenEdit_AssignUnitToMission('Ukrainian Flanker #5', 'Aggressive Patrol')
ScenEdit_AssignUnitToMission('Ukrainian Flanker #6', 'Aggressive Patrol')
end

```

Depending on the roll, the six Su-27s will either stay grounded (NONE-assigned to no mission), launch but only stay near their air bases (assigned to the “Close Patrol” mission), or launch in a mission with a large prosecution area, making them more eager to fight (assigned to the “Aggressive Patrol” mission).

Another is changing sensor statuses to active or passive via **ScenEdit_SetEMCON**. The template is as follows:

```

ScenEdit_SetEMCON('EMCONSubjectType', 'EMCONSubjectNameOrID',
'EMCONSettings')

```

SubjectType can be a unit, group, mission, or side.

For a single unit it could be:

```

ScenEdit_SetEMCON('Unit', 'USS McCracken', 'Radar=Passive')

```

This turns off the radars on a unit called the USS McCracken.

Whereas for an entire side it could be:

```

ScenEdit_SetEMCON('Side', 'USSR', 'Sonar=Active;Radar=Passive')

```

This turns off the radars but turns on the active sonars for every “USSR” unit.

This can be combined with triggers (see 5.5.3, Triggers) to enhance scenarios in the following ways:

- Combined with time-related triggers to symbolize intermittent emissions. A unit doesn’t have to go to the extremes of either always or never using the active sensors.

- Combined with detection-related triggers to symbolize a unit commander going “Ok, the cat’s out of the bag, better emit for the sake of awareness” without needing to be explicitly under attack.
- Combined with special actions to symbolize a general order, particularly if it’s on an allied side the player otherwise doesn’t directly control.

Still another Lua capability is changing the weather. The **ScenEdit_SetWeather** function has four inputs: temperature (increments of 0 to 50 degrees Celsius), rainfall (increments of 0 to 50), clouds (increments of 0.1 to 1), and sea state (increments of 0 to 9).

So in basic terms, running the following **SetWeather** action-

```
ScenEdit_SetWeather((35),(12),(0.6),(4))
```

...will lead to a base temperature of 35 degrees Celcius (95 degrees Fahrenheit), moderate high clouds and rain, and a sea state of 4 (wind 11-16 knots, small waves).

Lua is capable of much more than these major examples. It can even create a type of “AI” if the programmer is skilled enough.

If in doubt about anything, or if one simply wants to learn more, ask online in the “Lua Legion” subforum at <http://www.matrixgames.com/forums/tt.asp?forumid=1681>.

5.4 Editor Drop-Down Menu

The editor drop-down menu only appears in ScenEdit mode and contains the scenario creation functions.

5.4.1 Scenario Times + Duration

This drop down allows you to set the current scenario time, the scenario start time, scenario duration, complexity, difficulty and location. The exact date of the scenario will change everything from national borders to the location of polar ice.

Scenario Current Time: Sets the current scenario time.

- Enter a date by either selecting and manipulating the individual elements of the date field (day name, Month, Day, Year) or clicking the calendar icon which will launch a small calendar dialog that can be used to choose a date.
- Enter a time by manually manipulating the hour, minute and second fields or typing them in.

Scenario starts on: Sets the time the scenario starts on.

- Enter a date by either selecting and manipulating the individual elements of the date field (day name, Month, Day, Year) or clicking the calendar icon which will launch a small calendar dialog that can be used to choose a date.
- Enter a time by manually manipulating the hour, minute and second fields or typing them in.

Scenario Duration (And lasts for) Field: Enter a numeric value in the days, hours, mins (minutes) fields to set this. This can be changed after playtesting if one feels the scenario is too short or too long for its own good.

Complexity: Allows editors to set a scenario complexity using a 1 to 5 scale. This will be reflected by the Complexity green bar scale in the load scenario dialog. Complexity is inherently arbitrary and up to the scenario creator, but a general rule of thumb is that scenarios with more units and/or a situation that requires more setup, such as aerial refuelling or coordinated time-on-target strikes, should have a higher complexity level.

Difficulty: Allows editor to set a scenario difficulty using a 1 to 5 scale. This will be reflected by the Difficulty green bar scale in the load scenario dialog. Like complexity, difficulty is inherently arbitrary and up to the scenario creator. Scenarios should be playtested to see the 'true' difficulty before a final score is assigned. However, a scenario where the win condition can be consistently achieved should get a low ranking, and one where it cannot easily be accomplished, even by the scenario creator, should be the opposite.

Location: Allows editor to fill in a text value noting location that will be displayed in the location field of the load scenario dialog. The location can be anything from geographical locations (I.e, GIUK Gap, East China Sea) to conflicts (Persian Gulf War, Indo-Pakistan War).

Title + Description: This dialog allows editors to add the scenario Title and descriptive text that appears in the Load Scenario Dialog.

Title Field: Enter a text value to give the scenario a title. It will appear in the load scenario dialog under title.

Description: This field allows editors to add text, pictures and links to the scenario description that appears in the load scenario dialog. It is copy and paste capable and includes most common Windows text edit functions.

To add text: Type and edit using the appropriate buttons. You can highlight over existing text and use the function buttons to make edits.

Pictures: Add pictures as needed.

Links: Add links to background materials, etc. as needed.

5.4.2 Database

Current: This displays the current database in use by the scenario.

Use Latest DB3000: This updates the database to the latest version of the DB3000, covering 1980 to the ever-shifting "near future". NOTE: Changing the DB in this fashion will delete all sides, so it's important for scenario designers to choose a DB *before* adding sides and units.

Use Latest CWDB: This updates the database to the latest version of the CWDB, covering 1946 to 1979.

Load Other DB File: This allows a database file to be manually loaded.

Upgrade Scenario To Latest DB Version: This allows an existing scenario to be updated to the latest version of its respective database without losing any units. It can be accomplished as either a “shallow” or “deep” rebuild, or used to update multiple scenarios. See 8.1, Scenarios Vs. Databases

5.4.3 Campaigns

This allows for the creation of campaigns of linked scenarios.

In it, the player can attach scenarios, attach attachments, and set a pass score to enable progress to the next mission in campaign play.

- Use **“Add scenario”** to select an individual scenario for part of the campaign. Likewise, **“Add attachment”** adds a scenario attachment.
- **“Remove selected”** is self-explanatory and removes the selected element.
- **“Move Up/Down”** changes the position of the selected element, useful for changing the order of scenarios.
- At the bottom of the campaign dialog, enter an appropriate pass score (what score the player needs to obtain in order to reach the next scenario) in the dialog box and press **“set”**. It will change the score for the selected scenario.
- **“Title and Description”** describes the beginning of the campaign, what players will see at the menu screen.
- **“Ending Text”** describes the ending of the campaign, what players will see after they’ve beaten it.
- **“Save Campaign”** saves the current campaign.

Create New: This opens up a blank campaign editor window where scenarios (not necessarily the currently loaded one) can be added.

Load From File: This loads a previously existing campaign for editing. Scenarios and attachments can be added or removed, the order and pass score can be changed, and so can the opening and ending text.

5.4.4 Add/Edit Sides

This function allows you to add sides to a game and edit their properties. Sides may be human or computer controlled. A side’s relationship with other sides is determined by its posture. Each side also has its doctrine which is part of the mechanism that governs its behavior.

- **Add :** Click **“Add”** and type a name for the side in the box that opens. A new side will be added to the scenario
- **Remove:** Select a side and click **“Remove”**. This deletes all units and missions for that side.
- **Rename:** Select a side and click **“Rename”**, Type in a new name in the box that opens. The side will be renamed.

Briefing: This opens a window to edit the briefing of the selected side.

Postures: This determines how each side interacts with other sides. The choices are Neutral, Friendly, Unfriendly, and Hostile. Select the side, then click **“Postures”**, in the box that opens, select the desired posture from the drop down menu. Posture choices:

- **Friendly:** Sides will not fire on each other and share contact information and air facilities.
- **Neutral:** Sides will not fire on each other and do not share sensor information.

- **Unfriendly:** Sides will not fire each other but will intercept when on patrol missions. They can conduct electronic warfare against each other.
- **Hostile:** Sides will fire on each other and can conduct electronic warfare against each other.

Doctrine–RoE button: This sets the default Doctrine, RoE, and EMCON for a side. All existing and Lua-spawned units will inherit these settings unless changed by a mission, script, or manual selection. The editor can set these to be editable by players in game mode by checking off the Player Editable checkboxes.

Side is Computer-Only: If checked, this makes the side inaccessible to the player in normal play. This is typically used to designate the OPFOR side in a scenario, and means that missions and tasks will have to be set for it before the scenario can be playable outside of ScenEdit.

Collective Responsibility: If checked, the actions of one unit in a side will be treated as reflecting the entire side. If unchecked, it will not be. For example, with **CR** checked, a unit from Side A being marked as hostile will cause all of Side A to be marked the same way, while designating that unit with CR unchecked will just change only that one unit.

Can Auto-Track Civilians: If checked, civilian and commercial units will automatically appear as visible to the side regardless of its awareness settings.

Awareness: This sets the side's level of awareness of the units in the scenario. Note that any unit may be set to be automatically detected and classified by the scenario author, regardless of this setting.

- **Normal:** All units must be detected and classified using suitable sensors.
- **Auto Side ID:** All units must be detected but the side identified upon detection.
- **Auto Side and Unit ID:** All units must be detected but the unit's side and unit type is provided upon detection.
- **Omniscient:** Similar to "God Mode" which reveals everything detectable.
- **Blind:** Nothing is revealed.

Proficiency Settings: This sets the default proficiency for a side. There are five proficiency levels ranging from 'Novice' to 'Ace', with 'Regular' as the default norm. The proficiency levels most visibly affect aircraft agility, with novice aircraft having only 30% of their nominal capability, veterans being able to equal the on-paper level exactly, and aces capable of even exceeding it.

In addition to aircraft agility, proficiency also affects damage control, ability to fly at low altitudes without advanced avionics, ability to withstand g-forces during hard maneuvers, reaction time (extremely useful for engaging fast moving aircraft or missiles), and the accuracy of unguided weapons.

Proficiency is inherently subjective and up to the player. There is no locking the player into select proficiencies for select countries. However, some possible guidelines are:

- **"Regular"** as the default and as a backup in case all else fails or a suitable choice cannot be found by the scenario author.
- **"Cadet"** as a nation with a poor historical record and/or a unit new to battle. **"Novice"** as an even more extreme version of **"Cadet"**.
- **"Veteran"** as a nation/unit with decisive combat experience and/or excellent training. **"Ace"** likewise as a similarly extreme side in relation to **"Veteran"** in the opposite way as **"novice"** is to **"cadet"**.

Individual unit proficiency can be changed from the side norm via either the right-click on unit dialogue (see Section 4.1) under "scenario editor" or in the order of battle menu (see Section 6.3.3).

Historical note: The unit proficiency symbolizes the disconnect between tactical and operational skill. Proficiency can swing marginal fights one way or another but is (generally) less important when one side or unit has a clear supremacy. Often, putting assets in the right place at the right time is more important than the exact performance of those assets.

Switch To: This drop-down allows you to switch the view perspective in the editor to each side you have created.

5.4.5 Edit Briefing

This function calls the edit briefing dialog which allows players to add text, pictures and links to the side briefings that appear for each side in the load scenario dialog and briefings dialog. You must switch sides using the editor switch to function to edit and add unique briefings for each side.

Briefings can be either formal with the style of an actual military document such as the traditional American “five paragraph order”, or informal. Whatever the format, they should include:

- The units under the player’s control.
- A less detailed description of units under enemy control (remember the fog of war).
- The objective and rules of the scenario.

The briefings should be able to lay out the basic guidelines for the scenario player, so that they do not have to wander around confused wondering what to do (if the objective is to hit a set of fuel tanks, the player shouldn’t waste time attacking air defense sites if they have limited resources, thus the briefing should mention that)

5.4.6 Edit Scoring Dialog

This dialog allows editors to apply scoring evaluation to their scenarios that appears in the victory conditions dialog. Points can be scored or lost in the game by using the scoring event in the event editor. This dialog is then used to evaluate that scoring.

To use:

1. Make sure the side you’d like to apply scoring to is selected in the Switch To dialog.
2. When the scoring dialog is open, enter high and low values in the Triumph and Disaster fields. Be aware that a campaign (see 5.4.3, Campaigns) will require reaching the set pass score to progress to the next scenario in it.
3. Make sure the numbers match the point total in the scoring events.

Note: Scoring overall is a hugely tricky situation. In many scenarios, a simple give and take of “destroy enemy units, gain points, lose friendly units, lose points” works fine.

But for others, this isn’t suitable. If the goal is to hit a certain high value target, that should take precedence over just trading blows. In extreme cases, hitting the target(s) alone can be the difference between success and failure. Hit it, get a triumph, lose it, get a disaster.

The scoring system thankfully allows for more nuance between the two extremes. Something like 1,000 points for hitting the target, -100 points for losing an aircraft, and 750 points being the line between minor victory and average allows for a bit of slack. Lose two aircraft, and the tradeoff for hitting the target is judged by the scenario designer to be worth it. Lose three and it’s a little too much. Miss the target and it’s an automatic failure.

The relative value of units should be considered by the scenario author. If the units are supposed to be expendable, then there should be little to no penalty for their losses. Likewise, an irreplaceable unit can justifiably result in an automatic loss for the entire scenario. The time period can be considered for losses of line units-if the side in question has plenty of aircraft to spare, a wave of replacements, and a politically high tolerance for casualties, accomplishing the objective beats taking losses. The reverse is also true.

Scoring can also be determined for pure gameplay reasons, even if it's not "historically accurate" per se. If the way to balance a scenario is to make one side's losses or kills worth a lopsided amount, that's a perfectly valid reason.

5.4.7 God's Eye View

Provides an omniscient view of all units on all sides in the game.

- This setting is useful for editors to see what is really going on while testing a scenario. Shifting from it to normal player views is used to provide an excellent contrast.
- This setting also allows editors to create strike missions for targets that have not been detected but still need to be specified.
- Keep in mind that this does not reflect what each player side sees at any time as they see things through often imprecise sensors.

Minefields: This drop-down menu allows editors to add and remove sea minefields.

To add and mines to a designated area:

1. Make sure you have switched to the side you would like deploy the mines.
2. Create and then select the reference points you would like to you use to define the area you'd like the mines to be placed.
3. Select Create minefield in designated area which launches the "add mines in designated area" dialog.
4. Choose the type you would like and how many (holding down the shift, control, or both keys while clicking will add more). Please consider the depth of the water and the depth parameters the mines can operate in prior to deploying. These are found in brackets after their name in the dialog. The different types of mines vary extensively in terms of both damage and sweepability, from basic moored contact mines to lethal rising mines.
5. When ready to add press the ADD button. The mines will be randomly distributed based on area and water depth.

To remove mines from a designated area:

1. Select the reference points or drop and select new reference points marking the area you'd like to remove mines from. Make sure that it covers the entire minefield.
2. Select remove mines from designated area in the Minefields drop down menu.

5.4.8 Weather

This function allows editors to manipulate weather and sea conditions.

Use this menu to manipulate weather conditions in the game. Simply slide the scales to manipulate average temperature, rainfall, sky (visibility) and wind/sea state conditions. These can be global or localized (see _____ Ref. Points Manager)

How weather impacts units in the game:

Temperature affects several sensors but primarily infrared ones. During the night, man-made and mechanical objects stand out more clearly against the background on infrared sensors. Conversely, during daytime (and particularly in hot climates) IR sensor range shrinks dramatically. Imaging sensors are less severely affected than older ones. Temperature differs from the “average” setting depending on region of the world-the equator is a lot warmer than the Arctic, obviously.

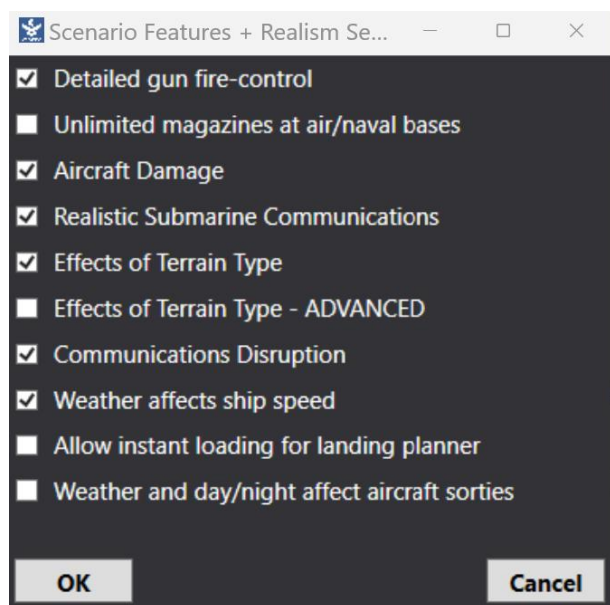
Rainfall affects both visual, IR and laser sensors. Visibility can shrink to as much as 1-5% of original range and similarly IR sensors are severely degraded by even light rain (e.g. in a rainstorm a Sidewinder may have an effective lock-on distance smaller than gun range).

Cloud coverage affects line-of-sight for visual and IR sensors. Light cloud coverage usually means scattered clouds at high altitude; aircraft can get under the cloud base and still visually acquire targets below, illuminate for LGBs etc. Thick cloud coverage means thick clouds at all altitudes; LGB guidance may be difficult or impossible unless the aircraft descends to low altitude, and spotting is likewise degraded.

Sea states affect primarily ship suitability (not currently modelled) and sonar performance. Surface ducting is useless in high waves and sonar ranges at or near the surface are severely degraded. High waves also raise the minimum safe altitude for low-flying aircraft. Sea state also affects the accuracy of naval gunfire, especially on lighter, more easily tossed vessels.

5.4.9 Scenario Features + Settings

This dialog allows editors to set default realism settings for scenarios. Some features are locked and require owning a certain DLC to unlock.



To select a setting just select a checkbox and click the apply button.

Detailed Gun Fire Control: When activated, guns have their Ph and CEP values realistically affected by the quality of their fire-control director and may altogether be incapable of firing if their sensors are blocked or knocked out. When disabled, guns can fire even without any director guiding them, and their accuracy is unaffected by sensor and weather limitations.

Unlimited magazines at air/naval bases: When activated, aircraft and ships now have unlimited ordnance at any port or airfield they deploy in. With this enabled, even if the magazine has one or zero specific missiles, a squadron of ten aircraft can all be deployed with a loadout of said missiles.

Aircraft Damage: When activated, aircraft are not immediately destroyed by any successful hit. Their survivability is highly dependent on both the weapon (light AAA is far weaker than a precise heavy SAM, and a modern fuze is better at inflicting damage than an older one) and the aircraft (Larger, multi-engined and/or more heavily armored aircraft are harder to destroy). A damaged aircraft that successfully returns to base will be down for repairs and thus take longer to ready than an undamaged one—often longer than the current length of the scenario.

When disabled, aircraft are destroyed in one hit by anything.

Realistic Submarine Communications: When activated, submarines diving below periscope depth will go out of contact with the player (see 10.7, comms disruption). They can be returned by right clicking on the ghosted “last known location” icon and selecting “summon to reestablish comms.”

An ELF signal amounting to “Ok, you’d better come up now” is sent, and, if the coast is clear with no immediate threats, the submarine returns to antenna depth, sharing any contact information it gained while deeply submerged.

Effects Of Terrain Type: When activated, this allows for different terrain types to have their effect on unit movement, visibility, and resistance to blast effects (see 3.1, The Globe Display for details). When deactivated, all types of terrain are treated similarly.

Effects of Terrain Type – ADVANCED: This option enables the effect of the skyline of the local terrain as well as the height of the mast of a unit to affect visibility. With this enabled, many older scenarios may be broken if the terrain wasn’t carefully checked prior to placement of units. Think of it terms of standing in the middle of a lot of skyscrapers with a pair of binoculars trying to see ‘through’ the buildings.

Communications Disruption: Enables comms disruption. See 10.7, Comms Disruption and Cyber Attacks.

Weather Affects Ship Speed: When activated, bad weather will slow ships down and in some cases even trap them in a near-stationary “heave”. Unsurprisingly, small ships are far more vulnerable to bad weather than large ones.

Allow instant loading for landing planner: This option allows for instant loading of units into vehicles when using the loading planner.

Weather and day/night affect aircraft sorties: Aircraft that are not suitably equipped (either built-in or through their loadout, e.g. LANTIRN pods) for flying at night or in bad weather, will not be able to take off. If

such aircraft are already airborne when unsuitable conditions occur, they run a high risk of crashing on landing. This feature can easily break older scenarios, so be careful when and how it is employed.

5.4.10 Merge Scenario

This fuses two scenarios into one. Scenario 1 acts as the ‘foundation’ which Scenario 2 is added to, and in the event of a conflict between a fundamental background characteristic like time or weather, Scenario 1 is used.

For DB match tolerance, the “exact DB option” (ie, DB3000 v477) is ideal. The others are very risky.

5.5 Events

Event Editor: The event editor is a powerful tool that allows editors to give more variability to Command’s Static AI and Editor’s scenario storylines by introducing conditional and/ or random events into the timeline of a scenario. In COMMAND terms, events are actions that occur as a result of a trigger.

You launch the event editor by clicking on Event Editor in the mission drop down and selecting Events from the dropdown. When you do so, the Event Editor dialog will launch the Events dialog.

Populated Event Dialog

The Events dialog includes a listing of all created events at the top and three function buttons at the bottom.

Events ✕

	Description	Is Repeatable	Is Active	Probability (%)	IsShown
▶	Game_Setup	☐	☒	100	☐
	Lualnit	☒	☒	100	☐
	Norway_UnitDestroyed	☒	☒	100	☐
	USSR_UnitDestroyed	☒	☒	100	☐

Create new eventClone selectedEdit selectedDelete selected

The event listing displays each event’s: description (name), if it is repeatable, if the event is active and the probability of the event occurring. These values are all set when creating the event.

The function buttons are:

- Create a new event: Launches the Edit Event dialog. See below for functionality.
- Edit selected event: Selecting an even from the list and pressing this button opens the event in the edit event dialog for editing.
- Delete selected Event: Selecting and event and pressing this button deletes it.

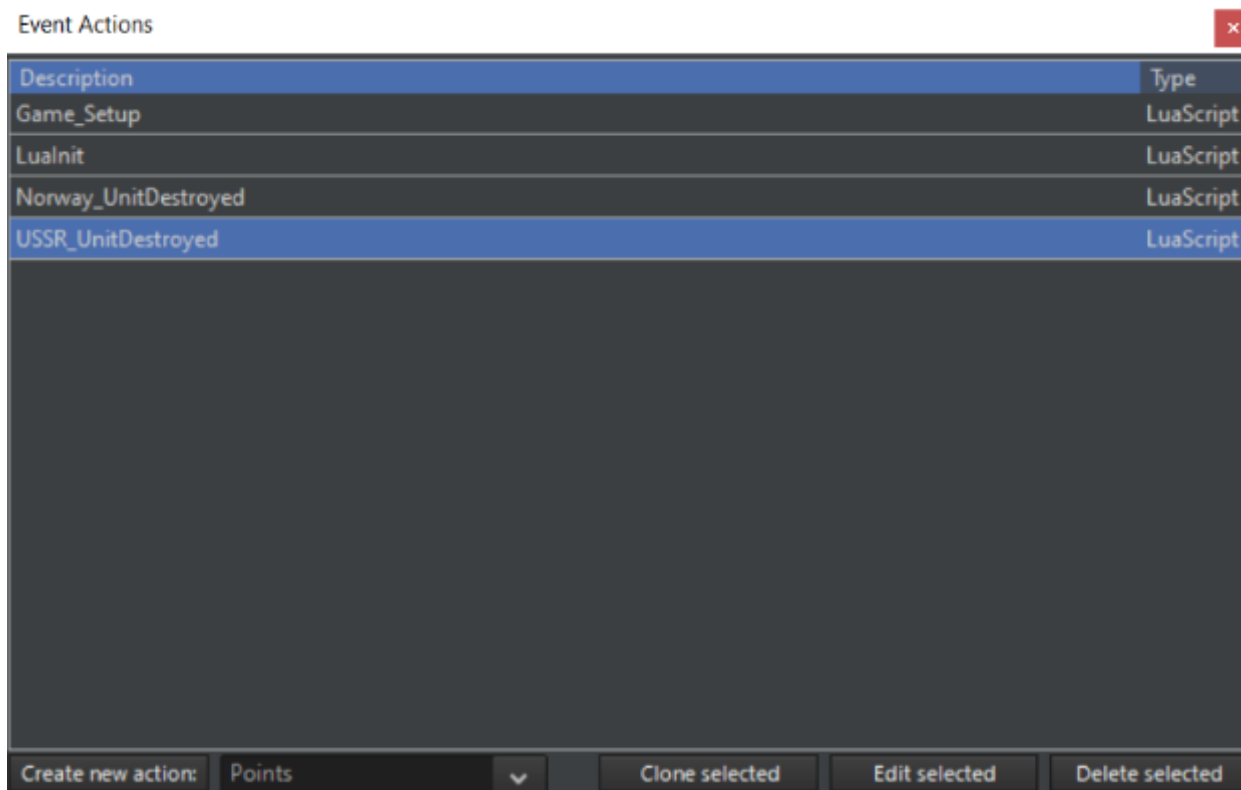
Creating an Event: As mentioned, events are actions that are the result of a trigger. So we'll work through creating an event while describing each dialog used.

First, click the **Create New Event** button which launches the Edit Event dialog. This dialog includes basic event parameters at the top and specific triggers and actions sections below.

Events are constructed from actions and triggers so it is best to build them first. You can do so by pressing the edit trigger or edit action buttons that are found at the bottom of their sections. You could also edit them independently from the event editor drop-down by selecting either Actions or Triggers. **Once you have constructed the necessary actions and triggers, you can begin construction of your event in the Edit Event dialog.**

5.5.1 Actions

You create actions by clicking on the edit actions button, which brings you Event Actions Dialog. This dialog has a listing of all created actions at the top and all function buttons at the bottom.



To create a new action you select the type from the drop-down menu at the bottom and press the Create New Action button which will launch the Edit Event action dialog, allowing you to construct the action and modify options and parameters. Descriptions of all are below.

To edit an existing action, select it from the list at the top and press the Edit Selected Action button, which opens the Edit Event Action dialog allowing you to modify the action options and parameters.

To delete an action, select it from the list at the top and press Delete Selected Action. Keep in mind that if you are using the action in an existing event it will be deleted as well.

What are Actions?

Actions are the result of a trigger, and may be used in many triggered events.

To setting up actions, click on Editor > Event Editor > Actions. A window opens listing the existing possible actions and options to create, edit, or delete the selected action.

Action Types:

Points: This allows adding or subtracting points from a side's score. Be aware that other points will be

- To use:
- Fill in a description of the action (ex. 100 Points to side Blue).
- Select a side that is to receive or lose the points.
- Add the Point change. Positive numbers to grant points and negative to subtract (ex. -100).

End Scenario: This action terminates the scenario. It is suggested that you add a message action prior to this letting the player know.

Teleport to Area: This is used to teleport units to within a reference designator defined area. This can be used to simulate the arrival of reinforcements.

To use:

- Make sure you drop some reference points to mark the area the units will be teleported to. Select/activate them and make sure other reference points are deselected. The units will appear at a random point in the area, so keep that in mind when making or selecting them.
- Launch the event editor and add action
- Type in a description (for example, "1st platoon lands at airport").
- Click on "add points currently highlighted" to set the teleport area.
- Select the units you would like moved. They are selected when highlighted in blue.
- Click the OK button.

Message: Creates a popup message with user-defined content. Please remember to tell players to activate their special message popup option in their Game Log message options to see this as a popup rather than a less visible entry in the game log.

To use:

- Fill in a description of the message (like "Blueside war warning").
- Use the drop-down to select the side that will see it.
- Use the text box to type the message you would like displayed.
- Click the OK button when complete.

Change Mission Status: Sets a mission to active or inactive. This action is designed to be used with the mission status field in the mission editor. Please be sure do set the mission status (active/inactive) to the status you'd like the mission to have a scenario start.

To use:

- Add a description of the action (such as “Activate Blue AAW Patrol”).
- First select a side from the Side drop-down and then a mission from mission drop-down.
- Click the New Status drop-down and select a status.
- Click OK.

Lua: Activates a Lua script. The Lua script must be manually entered to the action console by the scenario, although prepackaged template options are set for non-coders. For information on specific Lua scripts, see 5.3, Lua.

5.5.2 Special Actions

Special Actions have a Lua input. Mechanically, they are identical to “normal” Lua actions, but can be activated on cue by the player. For how to activate them in normal game mode, see 6.3.10, Special Actions.

Special Actions for this side

Name	Is Active	Is Repeatable
Abqaiq Patriot Strike	☒	☐
Al-Kharj Special Op	☒	☐
 Oil Infrastructure Sabotage	☒	☐
Request Ballistic Missiles.	☒	☐

Create new special action

Clone selected

Edit selected

Delete selected

Saboteurs are waiting in place to hit Saudi Arabia's oil infrastructure-pipelines, rigs, and the likes. The country is too big to effectively defend every last segment, but it also means that damage will be limited compared to "jackpot" targets.

For scenario creators, special actions come in two forms: Functional and tactical. “Functional” special actions exist for the sake of providing a service to the scenario player, like replaying special messages, delivering weather reports, deploying a unit (with no guarantees for its safety once it *is* deployed), and the like.

Tactical special actions are things like “attempt to use irregulars to destroy an enemy SAM site” or “request additional missiles from storage”. Tactical special actions should have either a sort of balance (ie, you get reinforcements, but so does your opponent) or a potential outcome that ends badly for the side triggering it to prevent the player from just constantly choosing them, knowing there’s no downside to pressing the button.

5.5.3 Triggers

Triggers initiate events. A trigger can be used to initiate multiple events; however, each event must be set up separately. Triggers are created by selecting the Edit Trigger button which launches the Event Trigger dialog.

Description	Type
400k tank lost	UnitDestroyed
750k tank lost	UnitDestroyed
F-14 Shot Down	UnitDestroyed
F-14AM Shot Down	UnitDestroyed
Iran Refinery Hit	UnitDestroyed
Iran Rig Hit	UnitDestroyed
Iranian Aircraft Lost	UnitDestroyed
Kharg Tank Hit	UnitDestroyed
Market Report	Time
Power Plant3	UnitDestroyed
PowerPlant1	UnitDestroyed
PowerPlant2	UnitDestroyed
Saudi Aircraft Lost	UnitDestroyed
Saudi Node Destroyed	UnitDestroyed
Saudi Oil Rig Destroyed	UnitDestroyed
Saudi Pipeline Destroyed	UnitDestroyed
Saudi Refinery Destroyed	UnitDestroyed

Create new trigger: ▼ Clone selected Edit selected Delete selected

1. To create a new trigger you would select the type from the drop-down menu at the bottom and press the **Create New Action** button which will launch the edit event trigger dialog allowing you to construct the trigger and modify options and parameters.
2. To edit an existing trigger, select it from the list at the top and press **Edit Selected Trigger**, which calls the edit event trigger dialog allowing you to modify the action options and parameters.
3. To delete a trigger select it from the list at the top and press Delete Selected Trigger. Keep in mind that if you are using the trigger in an existing event it will be deleted as well.

Trigger Types:

- **Unit Destroyed:** Trigger occurs when the specified unit is destroyed.
- **Unit Damaged:** Trigger occurs when a specified unit or class of units
- **Unit Enters Area:** Trigger occurs when the specified unit enters an area defined by reference points. An option exists where a window of time may be specified such that the unit must enter the area within that time window to cause the trigger to activate.
- **Unit Remains In Area:** Trigger occurs when the specified unit remains in an area defined by reference points for a specified amount of time. The time is reset if the specified unit leaves the area for any reason.
- **Time:** The trigger occurs when the game time reaches a specified time. An effect similar but not identical to "Scenario is Loaded" can be achieved by setting the event to trigger at a time one second after the scenario is set to start (ie, if the scenario is set to start at 3:00:00 UTC, the trigger is set for 3:00:01.)
- **Random Time:** The trigger occurs at a random time within a user set interval.
- **Unit Is Detected.** The trigger occurs when a certain unit is detected by a certain side. Options range from any aircraft/ship to a specific unit name being positively identified.
- **Side Points:** The trigger occurs when the designated side's points either exceed, matches, or falls under the specified number of points.

- **Scenario is loaded.** The trigger occurs the moment the scenario is loaded.
- **Scenario Ends.** The trigger occurs when the scenario ends.
- **Regular Time.** The trigger fires every set interval (a dialogue opens, and it can be anything from a second to a day.)

Triggers Associated with Units

The “**Unit Destroyed**”, “**Unit Damaged**”, “**Unit Enters Area**”, and “**Unit Remains In Area**” triggers can be linked to units in a hierarchal fashion from a type of unit down to a specific unit. These types, below, are set when the trigger is created:

TARGET SIDE: Designates the side the trigger will generate from.

TARGET TYPE: Designates the type of unit. If any of the designated type of unit experiences the trigger criteria, then the trigger is activated. Unit types are:

- Aircraft
- Surface Ship
- Submarine
- Land Facility
- Land Unit
- Weapon
- TARGET SUBTYPE [DDG]
- TARGET CLASS [Arleigh Burke Class Destroyer]
- SPECIFIC UNIT [USS Arleigh Burke]

Triggers Associated with an Area

Some triggers have an area associated with them which are defined with reference points. The reference points defining the area should be selected prior to creating the trigger that uses them. The reference points used to define the area can be changed in the trigger edit dialog box.

5.5.4 Setting up Triggers

Click on **Editor > Event Editor > Triggers**. The “**Event Triggers**” dialog box opens listing the existing triggers and options to create, edit, or delete Triggers. Scroll down the pull-down listing at the bottom of the box and pick a trigger type then click “**Create New Trigger**”. The “Edit Event Trigger” dialog box is displayed.

NOTE: Triggers with areas defined by reference points associated with them should have the reference points already selected when creating the trigger.

Unit Destroyed:

1. Type in a description for the trigger.
2. Select the Side and desired unit settings from the drop down listings. These should be done from the top down since the subsequent available selections often depend on what is selected above it. If the trigger is going to apply to all types then the subsequent selections should be left as “None”. The same is true if selecting a subtype or class.

Unit Damaged:

- Type in a description for the trigger.

- Select the Side and desired unit settings from the drop down listings. These should be done from the top down since the subsequent available selections often depend on what is selected above it. If the trigger is going to apply to all types then the subsequent selections should be left as “None”. The same is true if selecting a subtype or class.

Unit Enters Area: The reference points defining the area should be selected prior to opening the Event Triggers dialog box.

- Type in a description for the trigger.
- Select the side and desired unit settings from the drop-down listings. These should be done from the top down since the subsequent available selections often depend on what is selected above it. If the trigger is going to apply to all types then the subsequent selections should be left as “None”. The same is true if selecting a subtype or class.
- Designate the earliest time the trigger can occur. If the designated unit(s) enter the area prior to the designated earliest time, the trigger will not activate.
- Designate the latest time the trigger can occur. If the designated unit(s) enter the area after the designated latest time, the trigger will not activate.
- Modifier: Modifies the trigger to a “Unit Leaves Area” trigger if checked.
- Verify the correct reference points are assigned to the trigger. Reference points may be added later after creating the trigger, removing reference points may be done at this time. At least three reference points must always remain .
- Select OK to create the trigger.

Unit Remains In Area: The reference points defining the area should be selected prior to opening the Event Triggers dialog box.

1. Type in a description for the trigger.
2. Select the side and desired unit settings from the drop-down listings. These should be done from the top down since the subsequent available selections often depend on what is selected above it. If the trigger is going to apply to all types then the subsequent selections should be left as “None”. The same is true if selecting a subtype or class.
3. Designate the amount of time the unit must remain in the area. Note that if the unit exits the area, the timer will reset.
4. Verify the correct reference points are assigned to the trigger. Reference points may be added later after creating the trigger, removing reference points may be done at this time. At least three reference points must always remain .
5. Select OK to create the trigger.

Side Points: Type in a description for the trigger. This trigger may be used to set victory or defeat in a game.

1. Select the side the trigger applies to.
2. Enter the amount of points and select whether the trigger occurs when the accumulated points for that side either exceeds, reaches exactly, or falls under the specified points.
3. Verify the correct reference points are assigned to the trigger. Reference points may be added later after creating the trigger, removing reference points may be done at this time. At least three reference points must always remain .
4. Select OK to create the trigger.

Time: Type in a description for the trigger. Set the date and time the trigger should occur. This trigger occurs when the game clock meets the designated time. Select OK to create the trigger.

Random Time: Type in a description for the trigger.

1. Set the date/time range within which the trigger should occur.
2. Press the “Set Random Time” button.
3. Select OK to finish creating the trigger.

5.5.5 Constructing an Event

Once you have completed the Actions and Triggers necessary for your event to work, you can begin creation of the event itself in the Edit Event dialog.

1. Give the event a name by populating the Description field with text.
2. Select the Event is repeatable check box if you'd like the event to repeat.
3. Enter a 1-100 numeric value chance of the event occurring in the Probability field.
4. Next go to the trigger section at the top and select the trigger you created from the drop-down and press the Add trigger button next to it, which will populate the trigger list above.
5. Next, go down to the Actions section at the bottom, select the action you created from the drop-down list, and press the Add Action button, which will populate the Action List with your chosen action.
6. If you'd like to make any edits to the actions or the triggers you can press the Edit Action or Edit Triggers button in their related sections which will launch their edit dialogs.
7. If you'd like to remove a trigger or action select them in their lists and press the Remove Selected buttons in the appropriate section.
8. Press the OK button on the left to save your work and complete the process. Your new event should now appear in the events dialog.

5.6 Scenario Batch-Rebuilder (SBR)

This allows functions of the Scenario Batch Rebuilder to be accessed.

Generate Template: This generates a basic configuration template for the scenario. See 8.8, Scenario Config Template Files for more details.

Generate Delta Template: This generates an advanced delta template for the scenario, essential for deep rebuild database updates.

Apply SBR Script: This applies an existing SBR script to the scenario. See 8.9, Editing Scenario Config Files, for more details.

5.7 Unit Actions

This allows actions concerning individual units to be taken.

5.7.1 Add a Unit

This function allows users to add a unit to a scenario. It can also be called from the ctrl + right click drop down menu.

Add new unit ✕

Type: Facility ▼

Side: Iran ▼ Edit sides...

Name: Armored Plt (Chieftain Mk.5 MBT x3)

Filter by...

Class: armor Country: Iran ▼

SubType: None ▼

Hypothetical: Show all platforms, both real-life and hypothetical ▼

Custom GUID:

Platform	Country	From	Until	Hypo
Armored Plt (Chieftain Mk.5 MBT x3)...	Iran	1975	0	☐
Armored Plt (T-62 MBT x 3) -- Iran (...)	Iran	1983	0	☐
Armored Plt (T-72 MBT x 3) -- Iran (...)	Iran	1994	0	☐
Armored Plt (Type 69 MBT x 4) -- Ira...	Iran	1986	0	☐
Armored Plt (Zulfiqar MBT x 3) -- Ira...	Iran	1997	0	☐

OK Cancel

To add a unit:

- Make sure you have the side you want to add a unit to selected in the “**Switch**” to menu.
- Select **Add Unit** from the Editor drop down and click a point on the map you’d like to place the unit. Please be sure that the place you selected to drop the unit you would like matches terrain suitable for the type (ships in water, buildings on land).
- Choose Type using the Type drop down. Your options are aircraft, ships, submarine and facility. Notice when you do so that your pick list on the bottom of the list switches to match your chosen type.
- Side should be the side you are set to but you could change it with this drop down. If you wish to add a side on the fly click the Edit button which will launch the Add/Edit Side Dialog.
- The Name field allows you to name the unit by deleting the default text and typing it what you would like.
- There are filter options to assist you in finding units to add:
 - Filter By Class Field: Typing in this field filters items based on what you type.
 - Filter By Country Drop Down: You can filter by country by selecting a country from the drop down list.
 - Filter By Hypothetical. You can filter by whether or not the unit is a historical or hypothetical one by selecting one of the three options from the drop-down list.

- Filter by Name (Alphabetically) You can filter alphabetically by click on the name Platform column. This arranges units by name alphabetically from a to z or z to a.
- Filter by Country (alphabetically): You can filter alphabetically by country by clicking on the country at the top of its column. This will filter by country alphabetically from A to Z or Z to A.
- Filter by From Date: Selecting From at the top of this column allows you to order by in service date.
- Filter by Until Date: Selecting Until from the top of its column allows you to order by out of service date.

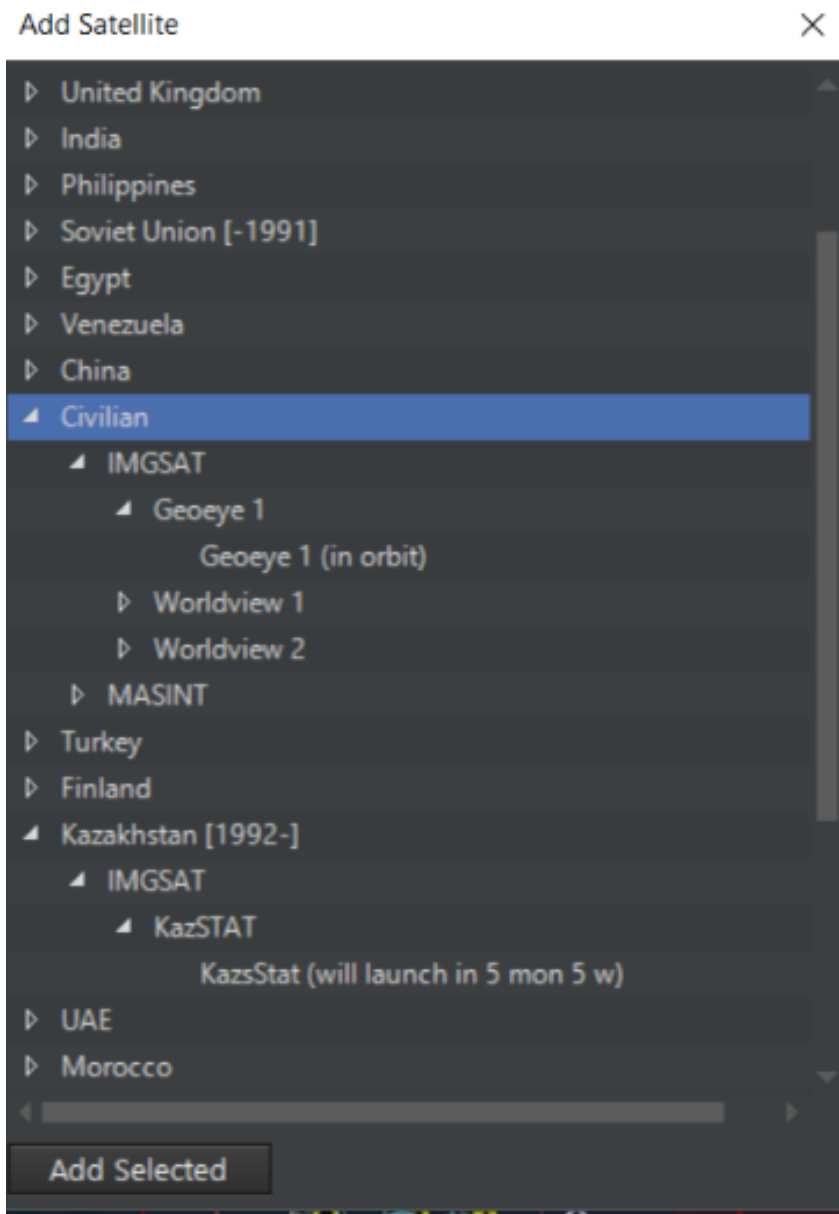
When you find the unit you would like to add select it by clicking on it and press the OK button. If it's a ship or facility it will be added to the map. If it's an aircraft, you will have to select a loadout and click ok.

The filter figures are there merely for reference and do not contain any limiters. Thus, the player can easily add in ahistorical units ranging from the plausible (Iowa battleships to a side called "US" in a mid-1990s extended Cold War scenario) to the impossible (F-22s to a side called "Bhutan" or "The Legendary Super Duper Force" in 1981).

This also enables units of the same type to be used as stand-ins for ones another country used but that are not in the database list for the country in question. For example, a Mozambican MiG-21bis (not in the database in the time of this writing) can be represented by an Angolan one.

5.7.2 Add Satellites

This menu allows users to add Satellites to scenarios. It is currently auto-filtered to only display satellites currently in orbit in relation to the current time value you have set.

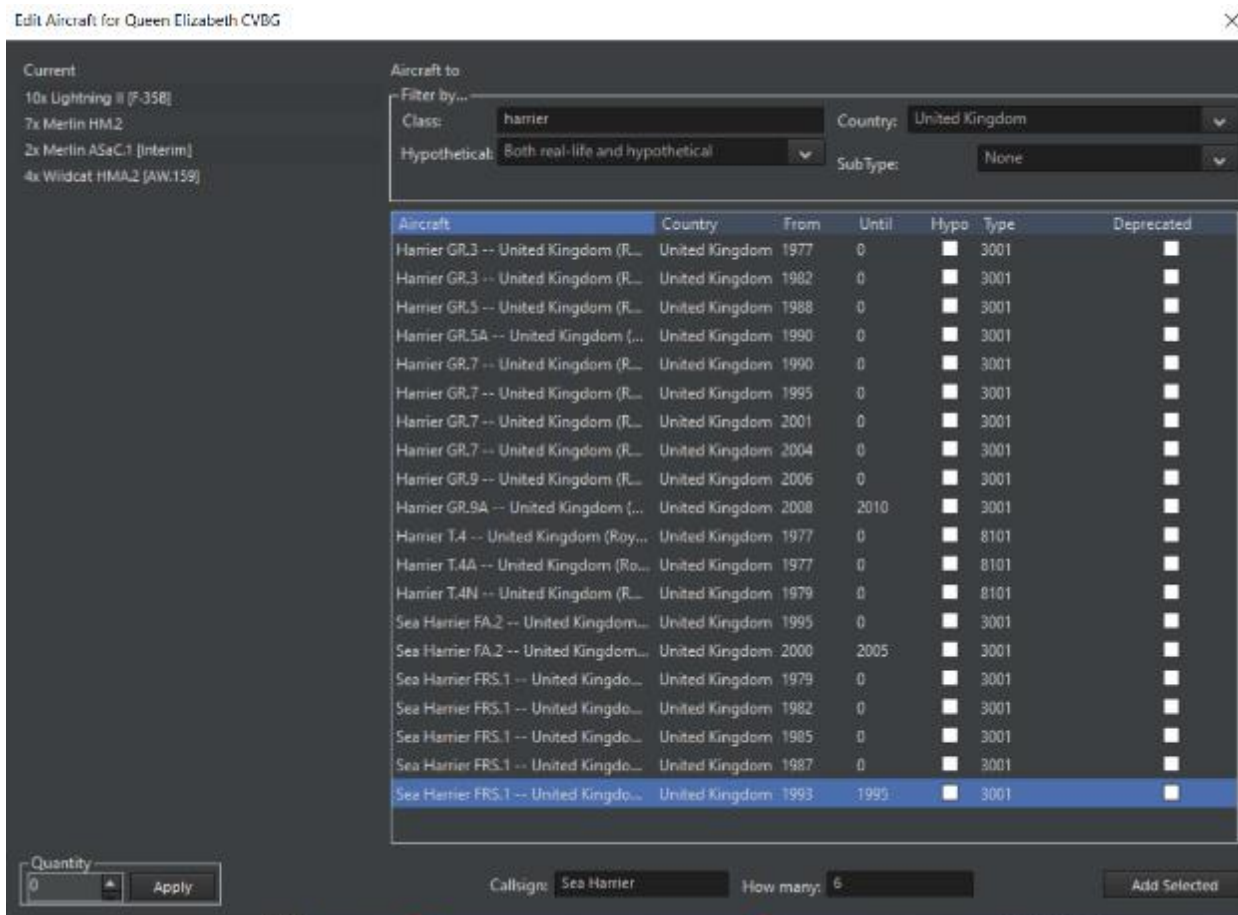


To Add a Satellite:

- Make sure the side you'd like to add them to are selected, and select **Add Satellite** from the editor drop-down, which will launch the **Add Satellite** dialog.
- To add a satellite to a scenario work through the nation, type, and model drill down lists and select the checkbox of the satellite you would like to add.
- Press the **Add Satellite** button when complete and it will appear on the GUI.

To Delete a Satellite:

Select the unit on the game map and press the D key to delete it.



5.7.3 Edit Aircraft

This function allows you to add aircraft to any facility or group that can accommodate them by calling the Edit Aircraft dialog. The current inventory of aircraft hosted on a unit are displayed on the left and the edit aircraft functions and pick list are on the right.

How to add an aircraft to unit or group:

1. Select the unit, group, facility or air base you would like to add aircraft too and select Edit AC in the editor drop down which will launch the Edit AC dialog. Please consider the amount of storage the unit has and runway lengths. The game does filter as best as it can but there are hybrid units that don't always fit the mold.
2. As in the Add Unit dialog you can filter by: Class Name, Country, Aircraft Name, Aircraft Country, From Date and Unit Date.
3. Select the aircraft that would like by selecting on it in the unit listing.
4. Callsign: This is an editable text field that allows you to give nicknames to units. The callsigns can be purely descriptive (ie "40th Fighter Regiment #1, #2, etc...") or more whimsical nicknames (ie, a group of F-15Js going by the callsign Tengu #1, #2). To avoid confusion, no two "batches" of aircraft should have the same callsign, especially if they stage from different bases.
5. How Many: Enter a numeric value for the number of aircraft you would like to add.
6. Press the Add Selected button when complete.

Removing aircraft from a group or unit

1. Select the units you would like to remove from the current inventory list on the right. When you do so notice that Remove Selected button has appeared.
2. Set the number of units you would like to remove in the quantity field and click apply.
3. Click the Remove Selected buttons to remove the unit

5.7.3 Add/Removing a boat or submarine:

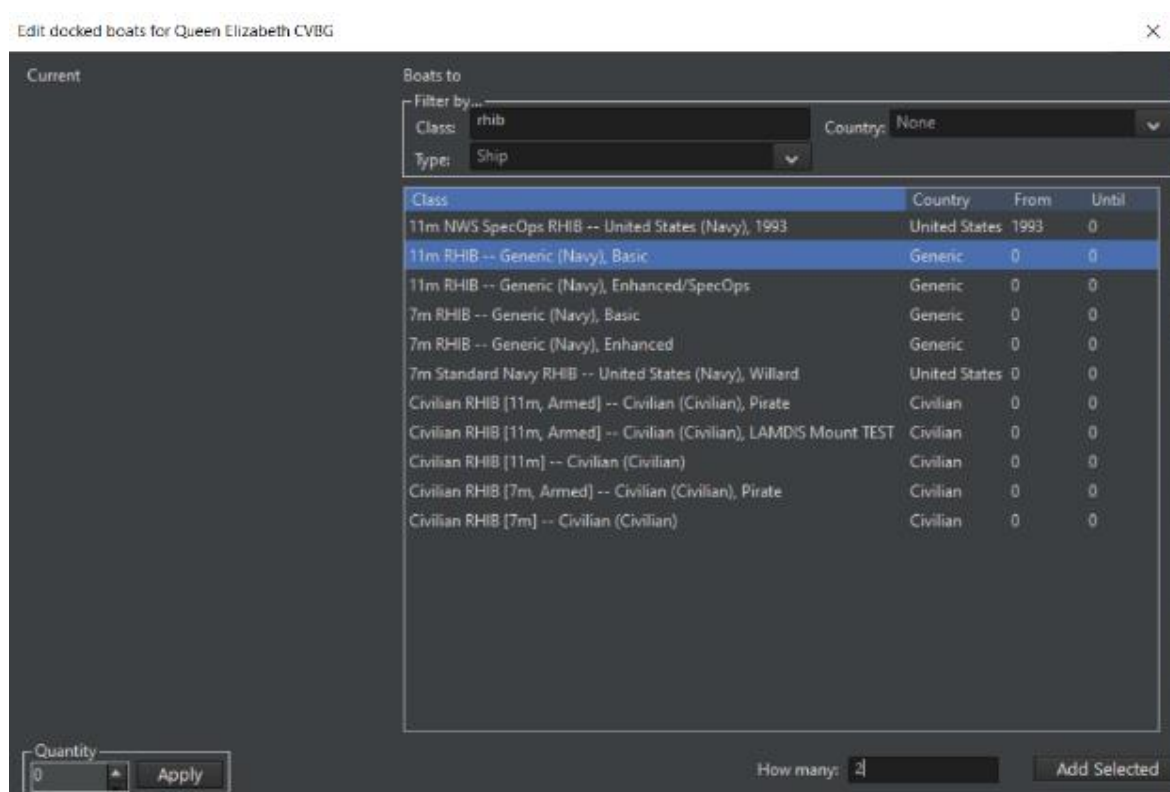
1. Select the vessel you would like to host the boat or submarine and select Edit Docked Boats from the editor drop down which launches the edit docked boats dialog.
2. Use the type, class, country filters to display a list of possible units to host.
3. Select the unit you would like by selecting it in the pick list.
4. Add the number you wish to add in the How Many field.
5. Click the Add Selected button to add them. They should now appear in the current inventory list on the right.

To remove a boat or submarine:

1. Select the units you would like to remove from the current inventory list on the right. When you do so notice that Remove Selected button has appeared.
2. Set the number of units you would like to remove in the quantity field and click apply.
3. Click the Remove Selected buttons to remove the unit

5.7.4 Edit Docked Boats

This function allows editors to add boats or submersibles/UUV's to facilities or boats capable of hosting them. When launched the Edit Docked Boats dialog displays the current inventory of units already hosted on the vessel on the left and filters and a picklist to add boats on the right. This list is already filtered to display ships that fall within the correct size limitations to be hosted.

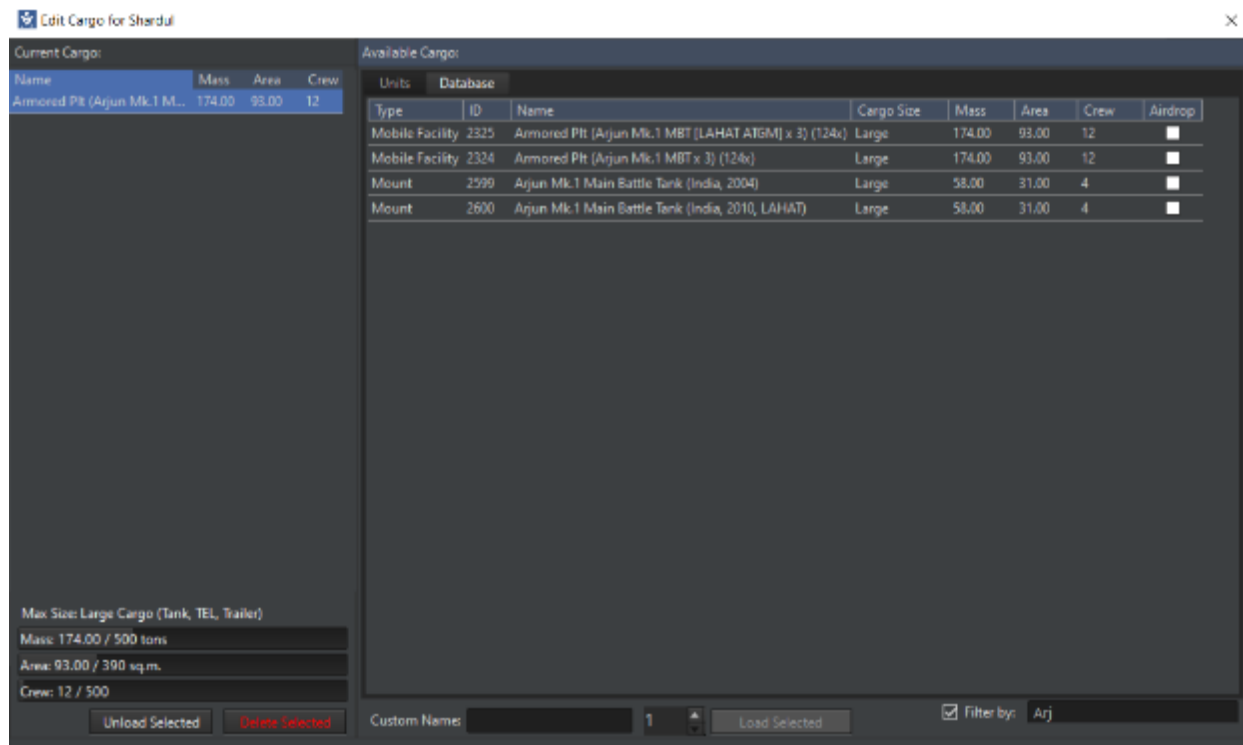


5.7.5 Remove All Units of the Side

Deletes all units for selected side. Be careful as there is no undo.

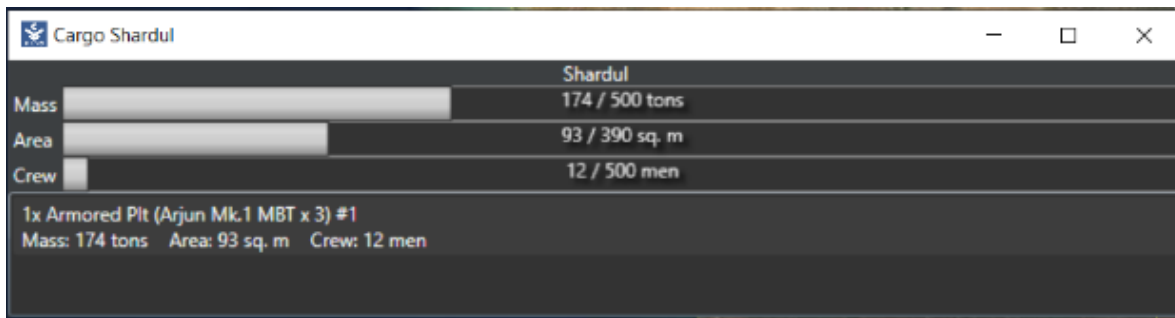
5.7.6 Edit Cargo

This function allows you to add cargo to a suitable unit.



To load an appropriate unit with cargo:

- 1: Check the DB viewer to see if it contains a cargo capacity. Cargo is limited by mass, area, and crew-a giant LST or Il-76 can hold what a C-130 or small boat cannot. Edit Cargo can also be used on airbases.
- 2: Click on the unit, then go to Editor/Unit Actions/Edit Cargo
- 3: Select the mobile land unit to add in the "database", then go to the bottom left corner. Use the numeric counter to determine the quantity of the unit, then press "Load Selected".
- 4: Press "OK" in the bottom right corner to save the changes.
- 5: The "Cargo" button should appear on the unit sidebar. Clicking on it reveals how much mass, area, and crew the parent unit is holding, as how much spare capacity it has left.
- 6: Repeat as necessary.
- 7: For air cargo, make sure to both add suitable cargo to the airbase, and load specifically desired ones onto the individual aircraft if necessary.



5.7.7 Other editor functions

Copy Unit: Many units of the same type and load-out may be placed by selecting an existing unit and pressing the hotkey “C” then clicking on the map to place a copy of the selected unit.

Clone Unit: As with “Copy Unit”, only Shift+C is used to make a more exact “clone” of the unit in question.

Move Unit: To move a unit on the map, select the unit then press the hotkey “M” then click on the map the desired new location for the unit.

Rename Unit: Selecting this or pressing “R” after selecting a unit opens a dialog where the unit can be renamed.

Delete Unit: Select the unit or group and press the “DEL” key.

Set Fuel QTY and Airborne Time: Accessible for air units, this allows their fuel quality and time airborne (for the sake of things like quick turnaround calcs) at the start of the scenario to be set. An “automatically adjust fuel” option can be checked, which sets it at the optimum amount for the current altitude plus a 10% reserve.

Unit is auto-detectable: If checked, the unit can be seen without being in range of a sensor. This is generally off by default for mobile units and on for fixed units such as buildings.

Set Orientation: Allows you to adjust the unit facing. This is crucial for land units that have sensors with a limited arc like the radars on Patriot missile batteries. To use, simply move the slider bar to match your desired bearing.

Hold Position: This instructs a selected land unit to stop and stay in its current position.

Remove All Units (Of This Side): This deletes all units in the current side.

Remove All Contacts (Of This Side): This deletes all contacts visible by the current side. It does not remove the actual units, only the contacts.

Scenario Attachments: This allows overlays, Lua scripts, local videos/sound, or installation imports to be attached to a scenario or removed. The attachments can then be used via a “ScenEdit_UseAttachment” Lua script.

5.7.8 Publish Scenario to Steam Workshop (Steam only)

Obviously only accessible in the Steam version of COMMAND, this allows a scenario to be placed in the Steam Workshop. The process is very simple.

- Open the desired scenario in ScenEdit mode and ensure it’s in a satisfactory condition.
- Select the “Publish Scenario to Steam Workshop” option from the editor drop-down menu.

From here, either an external preview image can be loaded via “Select Preview Image”, or an in-game screenshot of the current camera can be used via “Use Screenshot”. Once the preview image is determined, simply press Publish New Item and the scenario will be uploaded.

Other players with the Steam version can obtain the scenario by accessing the COMMAND Steam Workshop, locating the scenario, and pressing the subscribe button. A download/update will ensue, and the next time the Steam version of COMMAND is launched, the scenario will appear in a special tab on the scenario load dialog (see 2.2.1, “Starting COMMAND”) for easy access.

Published scenarios can be updated or changed by pressing the “Update Existing Workshop Item” button. The changed scenario will then be transmitted to all subscribers.

Package Scenario For Distribution: This bundles the scenario and all selected attachments together in one zip file for easy identification and loading by end users.

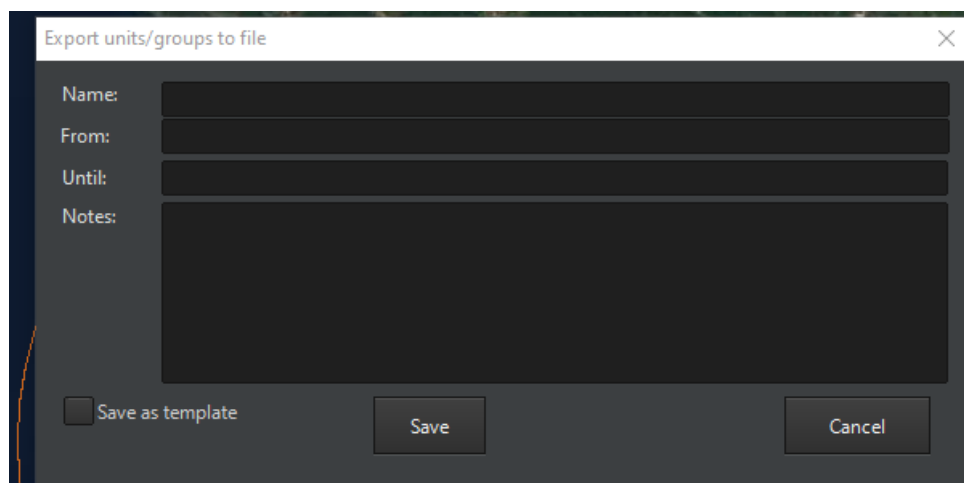
5.8 Import/Export Units

The import/export unit function is one of the most powerful tools in the game. It allows editors to save/export selected units and groups to a file and then import them into other scenarios at a later date or shared with other edits.

We have already prepopulated thousands of these files so that many of the airbases, air defense system or historical battlegroups might actually be there – saving you time and effort.

Save selected units/groups to a file - Creating import files

1. Select a unit or drag select multiple units and groups you would like to save so they are highlighted and then select save selected units/groups to a file from the editor>import/export units drop-down. This will launch the Export units/groups to a file dialog.
2. Fill in the Name, Valid From, Valid Until and Notes text fields if you'd like.
3. Press the Save button which will launch the Save As dialog.
 1. Give the file a meaningful text name and when press the Save button. By default it will be given an .inst file extension and saved to the import/export folder in your command directory.



It is now possible in COMMAND to save a group of units as a template, this allows you to place several copies of it when importing it without sharing the same GUID numbers. This is very useful to create a baseline formation and then use the import feature to create several copies of the same formation.

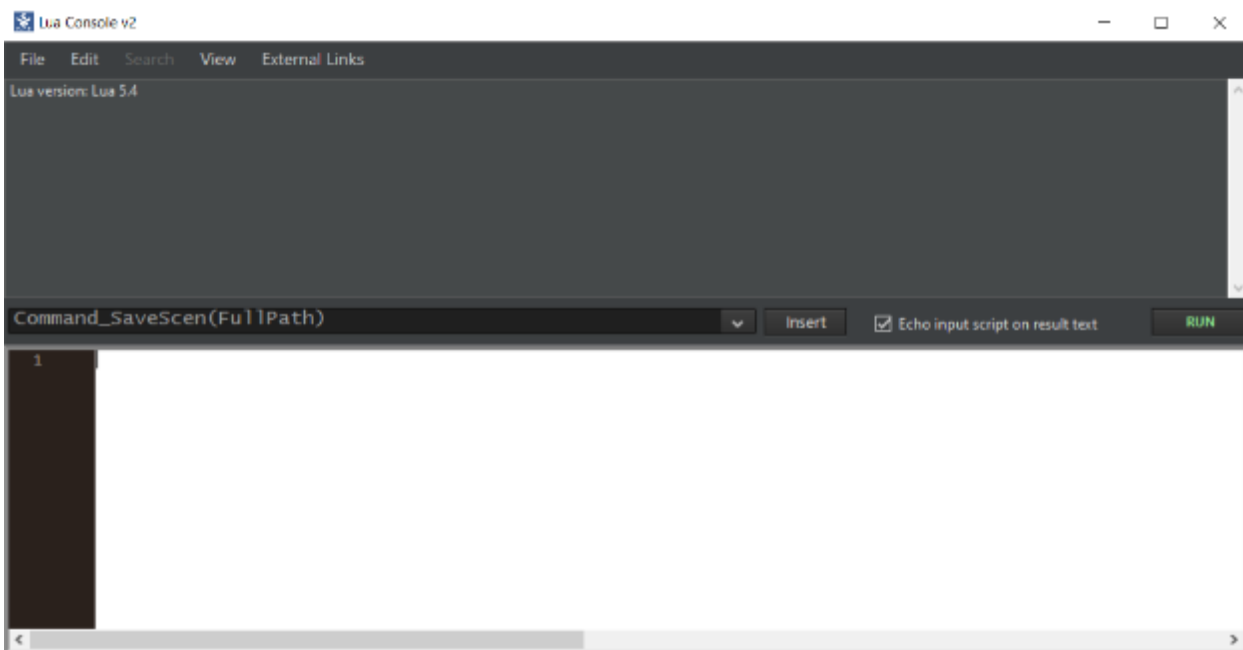
Load units/Groups from File - Importing

Selecting load units/groups from file editor>import/export units drop-down will launch the Load Group(s) from File dialog. The left side is a hierarchal listing of all folders and files in the import/export directory. The right side displays the information you entered when you created the file and a list of all group members.

To add units to your scenario:

- First find the unit or group you would like to add from the import listing on the left. You can open up the lists by clicking the + signs next to them to expose the files within folders.
 - You must be sure to match the import with the database you are using. We have marked all folders that contain coldwar database files as CWDB and then their names. All others are DB3k files.
 - You must be wary of duplicates as well. It is likely that the creator of the 1967 SAM site File used it to build the 1973 SAM site file and if you add both you'll get a duplicate key error message. If for some reason this is unavoidable just click okay and the duplicate will be skipped.
- Next mark the files you wish to import by selecting the open checkboxes next to them. If you'd like to select anything in a folder select the open checkbox next to it which will all select all folders and files under it.
- To import click the Load selected installation button which will populate the units for the selected side on the UI.
- To delete them you would have drag select and delete as you would a unit or group added any other way.

Lua Script Console: This opens a console where Lua scripts can be entered and run, either for the sake of testing or for the sake of scenario building (ie, using "ScenEdit_Addunit" to put a specific unit in a very specific place). Different parts are color coded to enhance readability.



Existing .lua files can be loaded into the console using the file/open key.

6.0 Drop-Down Menus

COMMAND'S menu and game controls are located at the top border of the UI.

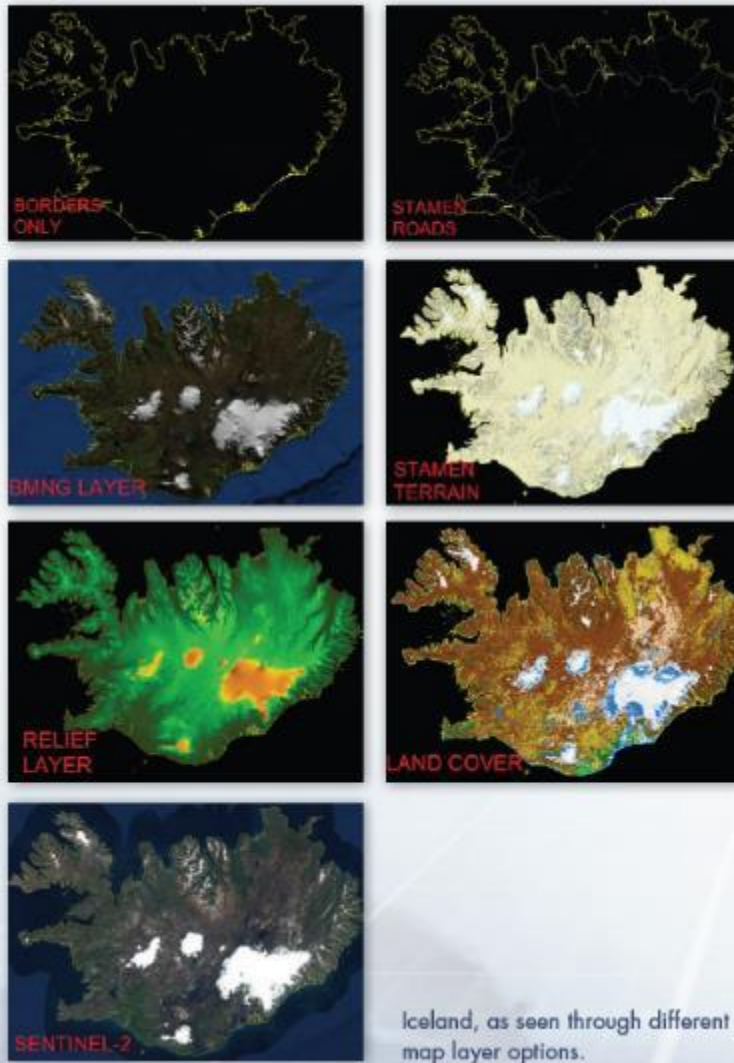
The menus are standard drop-down dialogs that are activated by clicking them once, which displays the drop-down menu's choices, and then clicking the item you would like to activate. There are a number of hot keys which call a number of these functions as well that you can view by clicking the Help menu item. You can find a list of all hotkeys at the end of this manual.

6.1 File

The "File" menu controls access to various game modes, similar to the start menu.

- Start Menu. This returns to the start menu.
- Create New Blank Scenario. This opens up ScenEdit mode and creates a new blank scenario without any sides or units.
- Load. This opens an existing scenario. If the current scenario is in normal play mode, the scenario will be opened in normal mode. If the current scenario/map is in ScenEdit mode, so will the opened scenario.
- Load Recent. This opens a list of recently loaded scenarios that can be re-accessed conveniently.
- Save. This saves the current scenario.
- Save As. This saves the current scenario under a potentially different name.
- Benchmark. This runs a chosen scenario in the "background" as fast as possible for a certain number of iterations. The time it takes to run the benchmark and pulse times of the simulation are shown. As the UI is not present and the player cannot influence the scenarios, a scenario with a large number of inbuilt AI missions on both sides is recommended for performance testing.
- Exit. This quits COMMAND.

COMMAND MODERN OPERATIONS



Iceland, as seen through different map layer options.

6.2 View.

The “View” drop-down menu controls the viewing of the map and certain display options.

- Toggle Full Screen/Windowed. This switches the map from full screen to windowed mode. It remains persistent across sessions of COMMAND, so exiting it in full-screen mode will lead to the next map opening in full screen mode, and vice versa.
- Zoom in/out. These options zoom the map in or out. The default is towards the center, but it can be changed by “zoom to mouse cursor” in _____, Game Options.
- Pan Map. These pan the map in any of the four compass directions.
- Select previous/next unit. This selects the next or previous unit in the scenario from the one currently highlighted.

- **Range/Bearing Tool.** This changes the cursor. Click on the map and then move the mouse cursor. A line will be drawn, and next to the cursor the distance (in nautical miles) and direction (in degrees) from the originally clicked point will be shown.
- **Message Log in Separate Window.** This moves the message log to a separate window from its default place overlaid on the screen.
- **Map Lat/Long Grid.** Displays a latitude and longitude layer that adjusts its increments to your zoom level.
- **Sentinel-2.** A high-resolution map layer. Enabled by default.
- **BMNG Layer.** When enabled, a satellite-terrain image of the world is shown. If disabled, the map appears as a yellow and black grid.
- **Relief + Bathymetry Layer.** Enables a colorful layer that displays elevations well. This is extremely useful for plotting terrain-masked attacks. It also shows a detailed view of underwater topography that can be used for various naval planning.
- **Borders and Coastline.** Enables and disables the yellow border and coastlines layer. The border and coastline layer is modified by year and reflects most border changes in the world during and after the Cold War. Yugoslavia and the USSR break into their successor states after 1991, while Germany reunifies.
- **Stamen-Terrain:** Adds a more detailed but graphics-intensive layer of the terrain map.
- **Stamen-Roads + Cities:** Similar to Stamen-Terrain, only highlighting roads and cities in detail.
- **OpenTopoMap:** Provides a view of the topography of a region using a detailed topographical map. This is useful for identifying key terrain features and planning strikes.
- **Land cover:** Displays the different types of terrain in clear detail. For more information see, 4.1, The Globe Display.
- **Custom Layers.** This adds and removes overlays.
- **Placenames.** This adds or removes the names of countries and cities on the map.
- **Day/Night Lighting.** When enabled, the map will appear lighter in areas of (local) day and darker in areas of night. When disabled, the map will stay the same no matter what the time.
- **3D View.** If Tacview is installed and enabled (see 10.8, Tacview), this creates a window containing a 3D depiction of the current scenario in Tacview. Note that Tacview will currently depict all units of all sides, similar to “God’s Eye Mode” being activated in Command.

6.3 Game

In the “Game” drop-down menu, pressing on the “Game Options” tab opens the game options menu with the following tabs:

6.3.1 Start/Stop

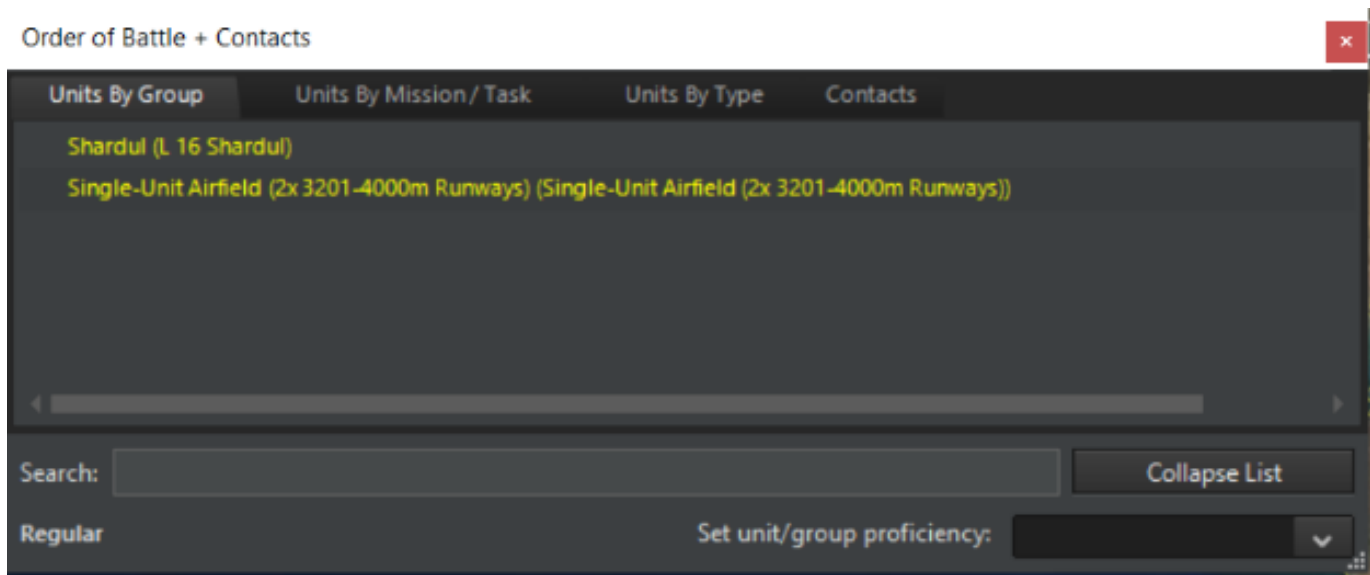
Under the “game” menu, this option depends on whether the simulation is running or paused. If running, the option to pause will be there, and vice versa.

6.3.2 Time Compression

These options increase, decrease, or reset time compression (the speed at which the scenario run). + increases time compressing (moving everything faster), - decreases it, and Enter returns it to its default (and slowest) speed at 1 second/real time.

6.3.3 Order of Battle

This dialog is a master list of units in the scenario and consists of three main tabs.



The first to be viewed upon opening the window is **"Units by Group"**. This contains a list of all units on the player's current side, sorted into groups and color-coded according to proficiency. Groups (from aircraft flights to naval/land groups to multi-unit airbases) can be retracted or expanded to reveal or hide their component and hosted units. The window refreshes as units are destroyed, change groups, or otherwise differ in status, making it useful for big/multiple monitor setups as a "reminder".

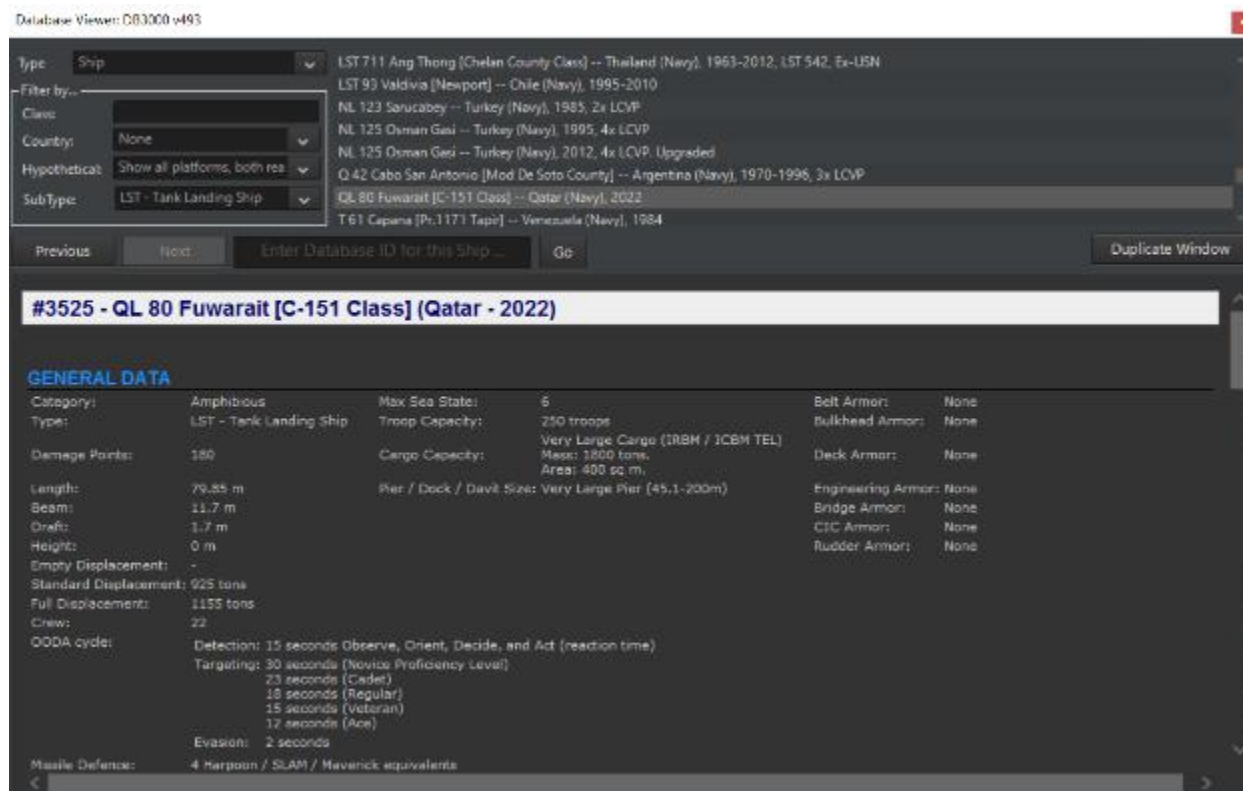
Clicking on any unit moves the camera to its location on the map. If the unit is hosted somewhere else, the camera moves there, ie clicking on an aircraft parked on the USS Nimitz will shift the camera to the carrier.

The second window/option is the **"Units By Mission/Task"** tab. This has the same units sorted by the missions they are currently assigned to. Unassigned units do not appear in this window.

The third is the **"Contacts"** tab. This displays all currently viewed contacts not belonging to the side in question, as well as their (color-coded) posture and status. As with friendly units, clicking on their names will shift the camera to the contact. The information in the window changes greatly with the data available. An unidentified surface ship contact will appear as just "Skunk #_", while a precisely identified contact will be something like "TAKR Admiral Kuznetsov, unfriendly, no damage, 4 Yak-141 Freestyles spotted hosted [on the flight deck]"

At the bottom of the Order of Battle screen, regardless of the current tab, are three items. The first is a search feature that will highlight anything (regardless of tab) containing the typed term. The second, in the lower left corner, is the set default proficiency of the player's side. The third, only accessible in ScenEdit, allows for the proficiency of highlighted units to be changed. This allows for mass proficiency changes without either changing the whole side or doing it one unit at a time and is the best way to change the proficiency of hosted aircraft.

6.3.4 Database Viewer



This dialog displays all units in the database that is currently selected (ie, DB3000 Build 494).

To view, select a type and unit from the list that appears on the upper right. Unit information is displayed below. This display also lets you filter results by name, country, or status as a real unit vs a hypothetical one, in addition to scrolling between chosen units and being able to find units based on their known database ID by typing it in. Finally, the DB window can be duplicated to allow for the viewing of different entries at once.

The filters can be combined. As an example, typing in “F-14” will result in all Tomcat variants in both American and Iranian service being shown. After “F-14” has been entered, selecting “United States” or “Iran” from the list shows only the ones from those specific countries. With “United States” selected, going to “Show Hypothetical Platforms Only” will narrow them down further to proposed upgrades like the AST-21 Super Tomcat that never actually saw service. With “Iran” selected, going to hypothetical only results in a blank display, as the country currently has no never-were F-14s in its database.

To reset the country filter, select “None” at the very top of the country list. To reset the hypothetical filter, select “show all platforms, both real-life and hypothetical”. To reset the name filter, simply delete the current name.

The “Weapons” section in the database viewer is not sorted by country, although it is separated by real and hypothetical units. It contains information on the range, signature, guidance system, and performance of weapons. At the bottom of every page in the weapons section is a list of all the platforms that carry it. This can vary considerably depending on the type of weapon. For instance, the AIM-54C Phoenix is only carried by several variants of American F-14 Tomcats, and the Fakour-90 version of the missile is only carried by one

single variant of the Iranian Tomcat. In contrast, the basic Mk82 500 pound bomb is carried by an almost countless amount of aircraft from many countries and time periods.

The “Sensors” section in the database viewer contains all current sensors in COMMAND and their default platforms. If the same sensor is carried by different types of units (ie both ships and aircraft), it will divide them into appropriate groups.

This display is important in helping make good decisions about how units and weapons should be used against certain targets in the game. Take note of ranges, weapon/sensor characteristics and damage point values. Sensor technology levels will affect their performance and ability to counter electronic interference.

Unit radar signatures can be altered by loadouts. Aircraft carrying external ordnance will have a greater radar signature, with stealth aircraft being dramatically affected. The database viewer displays the RCS of each loadout. Loadouts that are completely internal and thus do not affect RCS are highlighted in green.

Database Viewer: DB3000 v199

Type: Aircraft

Filter by...
Class: f-35
Country: None
Hypothetical: Show all platforms, both res
SubType: None

F-35A Lightning II -- Turkey (Air Force), Cancelled
F-35A Lightning II -- UAE (Air Force), 2025
F-35A Lightning II -- United States (Air Force), 2017, Block 3f, 7G Clearance
F-35A Lightning II -- United States (Air Force), 2019, SDB-I, JSOW
F-35A Lightning II -- United States (Air Force), 2021, SDB-II, JSOW
F-35A Lightning II -- United States (Air Force), 2024, SDB-II, JSOW
F-35A Lightning II -- United States (Air Force), 2025, 6x JATM, Block 3U/4, 9G Clearance
F-35B Lightning II -- Italy (Air Force), 2018

PreviousNextEnter Database ID for this Aircraft...GoDuplicate Window

Aircraft Loadouts

Name	Stores	Type	Front	Side	Rear	DB ID#
(Ferry) (Ferry)		Radar, A-D Band (30-2000 MHz)	-18.20 dBsm, 0.015 sq.m.	-16.00 dBsm, 0.025 sq.m.	-18.20 dBsm, 0.015 sq.m.	25639
		Radar, E-M Band (2-100 GHz)	-28.20 dBsm, 0.0015 sq.m.	-26.00 dBsm, 0.0025 sq.m.	-28.20 dBsm, 0.0015 sq.m.	
A/A: AIM-120C-5 AMRAAM, Internal Only (Air Superiority, BVR AAMs)	4x AIM-120C-5 AMRAAM P31.2	Radar, A-D Band (30-2000 MHz)	-18.20 dBsm, 0.015 sq.m.	-16.00 dBsm, 0.025 sq.m.	-18.20 dBsm, 0.015 sq.m.	25637
		Radar, E-M Band (2-100 GHz)	-28.20 dBsm, 0.0015 sq.m.	-26.00 dBsm, 0.0025 sq.m.	-28.20 dBsm, 0.0015 sq.m.	
GBU-32(V)2/B JDAM [Mk83], Internal Only (Standoff Strike, Land)	2x AIM-120C-5 AMRAAM P31.2 2x GBU-32(V)2/B JDAM [Mk83]	Radar, A-D Band (30-2000 MHz)	-18.20 dBsm, 0.015 sq.m.	-16.00 dBsm, 0.025 sq.m.	-18.20 dBsm, 0.015 sq.m.	25638
		Radar, E-M Band (2-100 GHz)	-28.20 dBsm, 0.0015 sq.m.	-26.00 dBsm, 0.0025 sq.m.	-28.20 dBsm, 0.0015 sq.m.	
GBU-49/B Paveway II, AIM-120C-7 (Standoff Strike, Land)	2x AIM-120C-7 AMRAAM P31.2 4x GBU-49/B Paveway II GPS/AGB [Mk82]	Radar, A-D Band (30-2000 MHz)	-6.30 dBsm, 0.23 sq.m.	-6.13 dBsm, 0.24 sq.m.	-6.30 dBsm, 0.23 sq.m.	30256
		Radar, E-M Band (2-100 GHz)	-6.56 dBsm, 0.22 sq.m.	-6.55 dBsm, 0.22 sq.m.	-6.56 dBsm, 0.22 sq.m.	

Loadouts for an F-35. The external loadout is far more detectable than the internal ones.

Clicking the “Duplicate Window” button will open up a second database viewer window. It will display the current unit by default but can be changed. This is useful for comparing the properties of different platforms.

OODA Cycle: The OODA (Observe, Orient, Decide, Act) cycle of a unit determines how fast it can respond to changing situations, especially dangerous ones. It is determined by both mechanical factors (representing the sophistication of automation on the unit), and human ones (representing crew skill.)

Thus, for targeting, even if a unit sees/detects the opposition, its OODA “clock” must tick down to zero from its stated beginning before it actually engages. An old, crude unit where everything is manually operated will have an OODA cycle far slower than a hyper-automated futuristic warship.

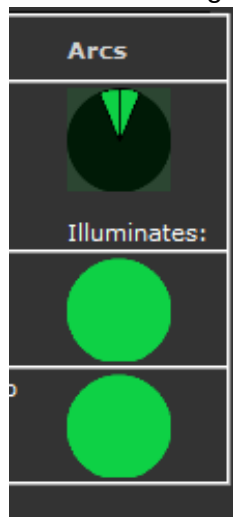
Besides the crew quality, the Combat System (shown in the database viewer as well) determines the practical reaction speed of a naval unit. Combat systems range from Gen 1 (World War II combat information centers, where contacts are manually marked on a transparent sheet), to Gen 7 (science fiction super-processors).

Agility: The agility of the aircraft determines its turning radius and ability to dodge attacks.

- <=1: Early jets or slow, support aircraft (E-3, A-50, C-5, C-130, etc..).
- 1.5-2.5: 1950s fighters, high-end bomber aircraft (B-1, Tu-160), medium attack aircraft (Su-24, F-111, A-6)
- 3-3.5: Third gen fighters. (MiG-21, F-4)
- 4-4.9: Fourth-gen fighters (F-14, MiG-29, F-16, Mirage 2000)
- 5+: Fifth gen fighters (F-22, Su-57, Eurofighter)

Note that units with the same “nameplate” can have different agility values. One example is the F-14, whose agility increases from 4 in the basic F-14A model to 4.5 in the upgraded, better-engined F-14B and F-14D models. Note that all types of units modelled in COMMAND have various levels of this targeting quality dependent on the generation and type of system. For an illustration of this, see the differences between an S-60 battery and a ZSU-57-2 mobile AAA. While both systems have the same base weapon, their response times are significantly different due to the nature of the layout and systems of the weapon.

Weapons and sensors have a unique arc where they can be pointed and utilized, this is represented by green circles on the right side where the individual sensors are displayed:



These arcs can also be turned on and off in the display options menu.

6.3.5 Browse Scenario Platforms

This dialog allows users to see all unit types used in the launched scenario. You can filter by side and the list is organized by platform type. Double clicking a unit name will launch the platform display where you can see information on the unit.

6.3.6 Scenario Description

This dialog displays a text description of the currently loaded scenario. The description is the same as the one appears on the menu, or when the scenario is started.

6.3.7 Side Briefing

This dialog displays the current selected side's briefing.

6.3.8 Side Doctrine/ROE/WRA/EMCON

This dialog lets you set the side's Rules of Engagement settings. Rules of Engagement settings are hierarchical in command and this is where you would set them as a player at the highest level to let them flow down to mission and then groups/units.

Doctrine and Rules of Engagement: Governs platform behavior, or posture, towards other platforms in the scenario. The Scenario Designer sets the initial Doctrines and ROEs and may enable the player to edit one or all of these. Doctrine and ROE can be seen by clicking on the "Doctrine and ROE" button in the Unit Information Box.

For more information, see "Unit and Group Doctrine" under "Buttons and Windows".

EMCOM Tab

EMCON is the Emissions Condition of the platform. See "Unit and Group Doctrine" under "Buttons and Windows" for more information.

WRA Tab

This controls the weapon release authorization (see 3.3.12, "Doctrine") for the entire side. Thus, every weapon used by the side will be displayed.

Withdraw and Redeploy

This controls the withdraw/redeploy (see 3.3.12, "Doctrine") settings for the entire side.

6.3.9 Special Actions

Some scenarios may have special actions that the player can choose to activate at will. This selection opens the window containing what, if any, special actions the scenario has. Special actions can be minor, major, high risk or low risk, and can include anything from reinforcements of one unit to posture changes. They can also be repeatable or one-time, like other events.

EXAMPLE: A call for reinforcements might give the player more units, but the enemy more as well as they shift more reserves in from their homeland to counter them. A lesser special action is something like raising a communications buoy on a submarine or receiving a weather report. See 5.5.2, Special Actions for how to add them as a scenario designer/editor.

Satellite Pass Predictions

This selection prompts a click on the map. Once a point on the map is selected, it calculates the approximate time when the satellites in the scenario will reach the point (the “enters coverage at” column) and the length of time they will remain there (the “dwell time” column). The satellites on all sides in the scenario are displayed, be they friendly or enemy.

Learning when a satellite would pass overhead and taking countermeasures was a well-practiced art throughout the Cold War, and the tool enables it to be practiced in COMMAND.

6.3.10 Recorder

This selection launches the game’s replay viewer. COMMAND has in-game recording capability as well as the ability to jump in and play at any point of that recording.

How it works:

You can start and stop any recording by clicking the **red recording button** at the top of the UI. When you do the recording is saved as a .rec file in the recording’s directory of your COMMAND directory.

These files are huge, given very long names and are not currently convertible to any other format. If you use this function often best practice is to navigate to the folder to rename the file to something that makes sense while retaining the .rec extension. You will want to clean this directory out as well as the files are huge. To view and manage any recordings you can launch the viewer by selecting recorder in the Game drop-down menu.

The replay viewer functions are very similar to any other video player’s control. You use the Load and Load most recent functions to pull of the videos you have recorded. You use all the video function buttons and slider to navigate to the point of the video you’d like to view. If you’d like to play you simply close the viewer and proceed to play.

6.3.11 Message Log

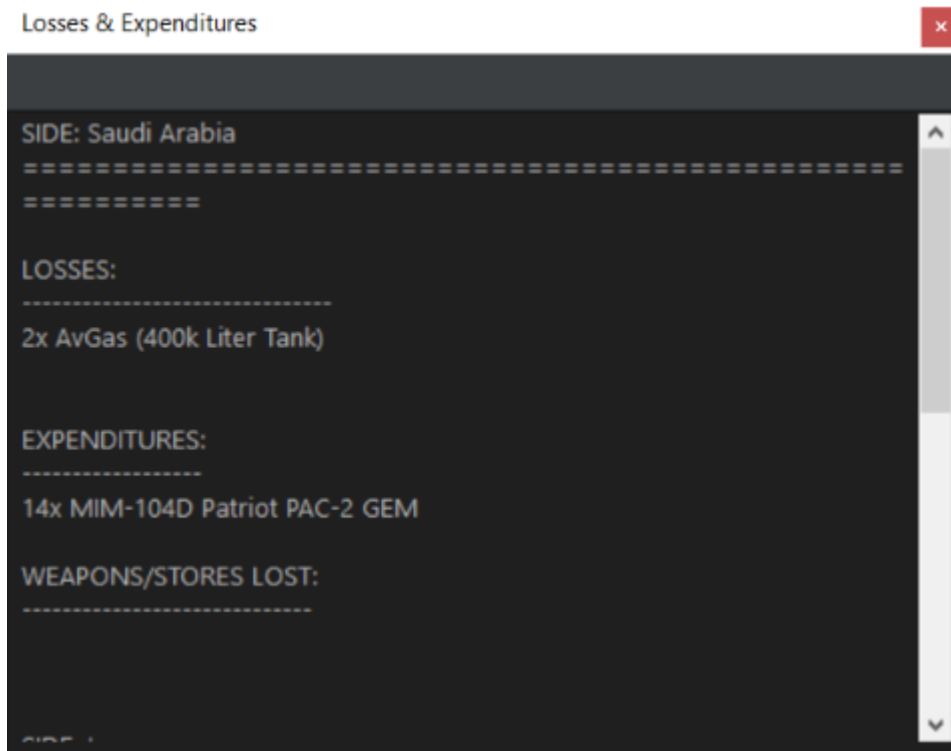
Select this menu item gives a drop down that gives players two options concerning their message logs, clear and print to file.

Clear: This function clears the currently selected side’s message log. This is useful for scenario editors to clear any messages that appeared during editing (God’s Eye View Mode). Scenario makers should clear the message log before the final versions are saved and released for public consumption.

Print to File: This function writes the contents of your message log to the AALog.Txt file your Command directory. This is good for after action analysis of the game mechanics.

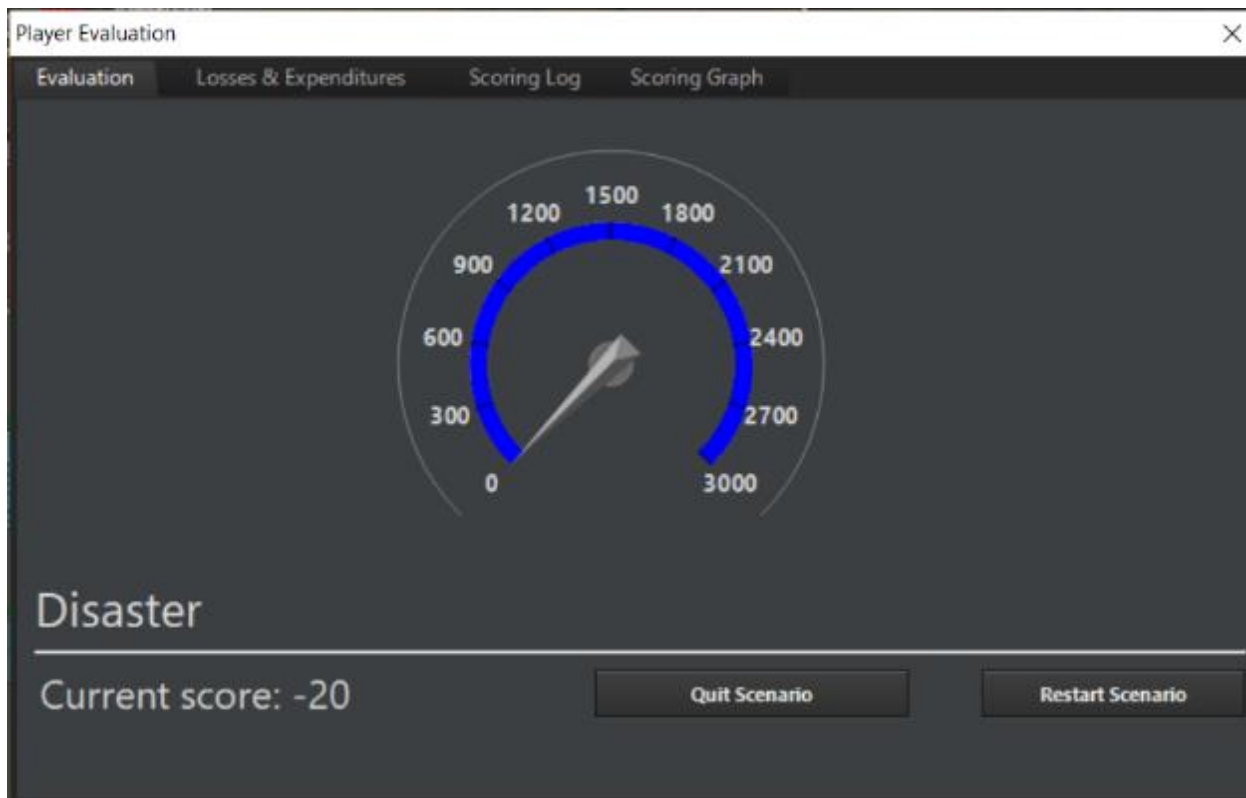
6.3.12 Losses and Expenditures

This menu item launches the losses and expenditure dialog which displays the losses and weapon expenditures of all sides of a scenario.



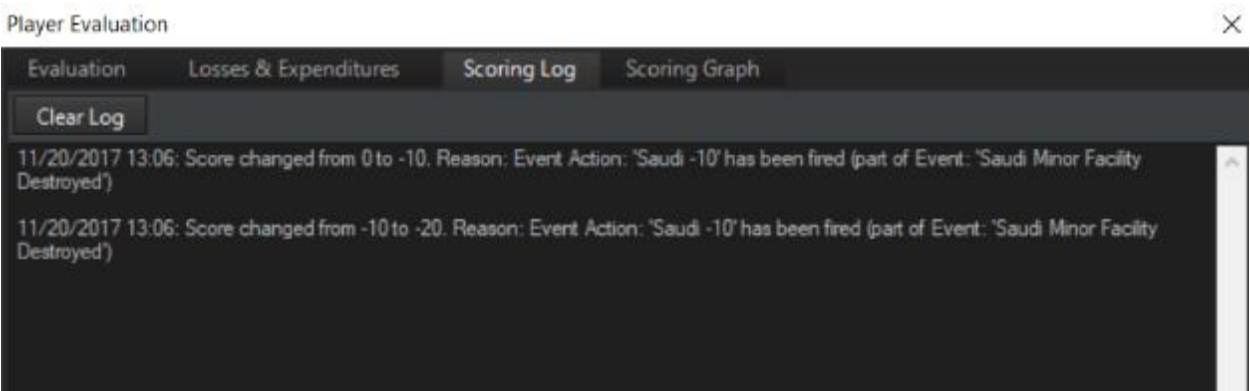
6.3.13 Scoring

Selecting this menu item displays the players score. Note that some scenarios do not have scoring enabled, whether deliberately or by omission.

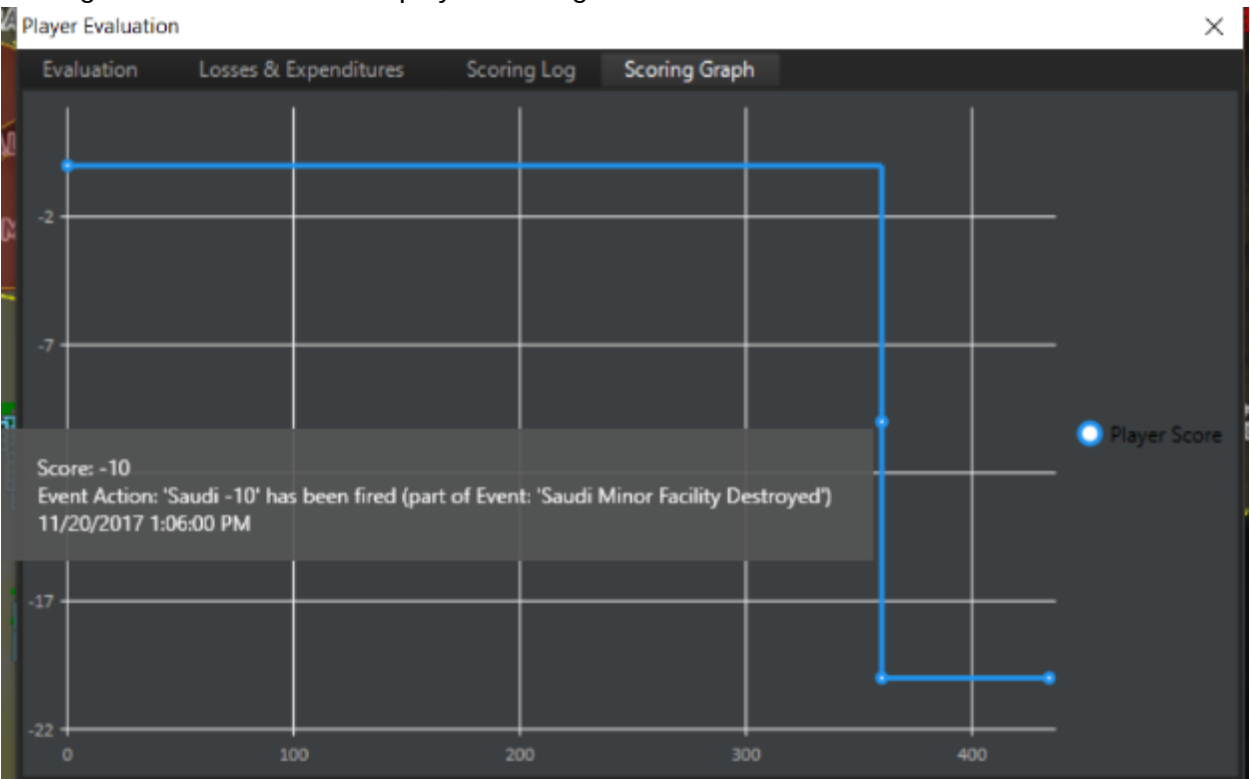


This dialog includes an **Evaluation**, a **Losses and Expenditures** tab, a **Scoring Log** tab, and a **Scoring Graph** tab.

The player evaluation tab displays scoring information. There is a “quit scenario” button at the bottom which allow players to end the scenario they are currently running. On the gauge, red represents levels of “defeat”, grey represents “average”, and blue represents levels of “victory”.
The Losses and Expenditures tab displays the unit losses and weapon expenditures of all sides of a scenario.



The Scoring Log tab illustrates the changes in scoring, for better and worse. It is helpful for reviewing if a scoring event fired without the player noticing in the confusion of battle.



The scoring graph shows scoring changes for the player’s side over time. Hovering the cursor over a specific point displays the change and the reason for it.

6.4 Game Options Window

6.4.1 General

- **Use Autosave:** COMMAND is saved periodically as “Autosave.Scen” in the “/scenarios” directory”. This should be used unless it places an undue strain on a lower-end computer. **Autosave automatically creates a different file for each scenario to prevent accidentally overwriting an existing autosave.**
- **Message Log In Separate Window.** This moves the message log into a different window from its default place overlaid on the screen.
- **Show altitude in feet.** This changes the altitude of units to be displayed in feet instead of meters.
- **Map zooms on mouse cursor.** If enabled, the map will zoom in on the mouse cursor. If disabled, it zooms in towards the center.
- **Hi-fidelity mode (in real time only).** If enabled, COMMAND will use “hi-fidelity mode” allowing for smoother movement on the lowest time compression setting.
- **Display unit status image.** If enabled, and if an appropriate image is in the database folder, the unit status bar will display a thumbnail image of the unit in question if it’s on the player’s side or a positively identified contact. If disabled, the image will not appear.
- **Allow switching to high-performance power scheme.** This allows COMMAND to let the computer switch to high-performance power systems even if it uses more electricity.
- **Show “Game Speed” button on toolbar in main window.** This provides a shortcut to the “game speed” options tab (see below) in the form of a red button on the toolbar.
- **Log debug information to file.** This logs debug information and is useful should a detailed bug report be needed.
- **DPI Scaling.** This changes the DPI scaling from either the computer’s internal defaults or a fixed “font” one at 125% scaling.
- **Fine-grained navigation**-this changes the settings of ships close to land.
- **Extra memory protection.** This, if necessary, provides extra memory safeguards to the computer.
- **Window Opening Behaviour** – This allows you to choose how the game should pause when various windows are opened during gameplay.
- **Enable Right Mouse Click move order** – This setting allows you to order units to move by a simple right click on the map instead of using the convention F3 key.
- **Indicators for aircraft contrails and ship/submarine wakes** – This option allows the display of contrail and wake status on the map under a unit’s datablock. This makes it easier for the player to be more aware that his units may be visually detected at significantly longer ranges because of these factors. These indicators can be disabled on the “Game Options” window (similar to noise & cavitation indicators) if they clutter the map unacceptably.

6.4.2 Map Display

- **Sonobuoy Display.** This makes sonobuoys either displayed with the same intensity as a normal unit, “ghosted” into appearing more transparent and faded, or not appearing at all.
- **Reference Point visibility.** This makes reference points appear either on their default size, appear smaller, or be invisible.
- **Map Cursor Databox Visibility.** This makes the black databox either appear next to the mouse cursor, at the bottom of the screen, or not be shown at all.
- **Map symbols.** This changes the symbols on the map to be either the default directional stylized symbols (large artistic symbols that rotate with their bearings and scale to each other’s size when

zoomed in), “stylized” symbols (a different type of artistic symbol that does not rotate), or “NDTS+NATO APP-6” (realistic tactical symbols)

- **Show Ghosted Group Members.** If “All groups” is selected, then the non-leader units of all friendly groups will be visible. If “Selected groups” is chosen, then they will appear only as the group icon centered on the group leader unless click-selected, and then the other members will show as ghosts. If “Do Not Show” is chosen, the non-leader units will never appear unless the game is switched to unit view.
- **Show Plotted Paths.** This shows the plotted courses for all units, selected units, or not at all, depending on the choice.
- **Show Diagnostics.** This shows internal game features such as pulse time and unit count on the top bar. It’s useful for bug reports to see (for example) exactly how much the sim has slowed down under certain circumstances.
- **Use Colored Datablocks.** this makes unit datablocks color-coded according to their posture. Blue for friendly, red for hostile, etc...
- **Use Personal Map Profile.** What this does is allow map display settings (as accessible from the “Map Settings” drop-down menu) to remain persistent across different scenarios. Pressing the “save current map profile as ‘personal’” button will identify the current map settings as the player’s “personal” ones, and checking the box will make those settings override the existing defaults, no matter what the scenario. Unchecking the box will revert the map settings to the defaults the scenario author has chosen. (WARNING: Scenario authors often set up the specific map settings to inform the player of vital information that may be missed if overridden. Use this at your own risk)

6.4.3 Message Log

The Message Log has four columns:

- **Message Type:** Regards the type of information that is displayed.
- **Show on Message Log:** Displays that message type when they occur if checked.
- **Raise Pop-Up:** If checked, initiates a pop-up message and pauses the game when the message type occurs.
- **Show Balloon:** This toggles a balloon appearing on the main map at the location of the message subject.
- **Timescale X1:** Immediately trigger the game to return from whatever time acceleration it was in to 1x.

6.4.4 Sounds and Music

This toggles in-game sounds or music on and off, and allows the volume of them to be changed if on. Turning music on plays all the current tracks in the [Installation folder]/Sound/Music directory. Music files can be manually added to or removed from there.

6.4.5 Game Speed

This enables certain features to be deselected in order to improve performance, particularly on lower-end computers.

6.4.6 TacView

Tacview: This allows the TacView executable path to be set.

6.4.7 Hover Info

Hover Info: This allows the contents of what comes up when the CTRL key is held down while hovering over a unit to be customized.

6.5 Map Settings Drop-Down Menu

Switch to Group/Unit View: This switches to the view of a group (grouped units are displayed with the group icon) or a unit (all units are displayed as individuals, regardless of their grouped status or not).

Show Ghosted Group Members: This changes the settings of ghosted group members, see “Map Display” above.

The Map settings drop-down allows players to manipulate what they see on the central globe display. They can turn features on and off by clicking on the drop down menu items. They are activated when a check mark appears next to them in the drop down.

Sensors Range Rings

- Air Sensors appear as white rings or arcs.
- Surface Sensors appear as yellow rings or arcs.
- Active Underwater Sensors appear bright green rings or arcs.

Weapons Range Rings

- Air weapons ranges appear as pink rings or arcs.
- Surface weapon ranges appears as dark red rings or arcs.
- Anti-land weapons ranges appear as brown rings or arcs.
- Underwater weapons ranges appear as dim green rings or arcs.
- Aircraft ranges (range-NOT radius) appear as blue rings.

Show Range Symbols For..

This allows range symbols to be toggle between visible for all friendly units, individual detected units, or not shown at all.

Show Non-friendly Range Symbols

When activated, the simulation will display the theoretical range-ring for a positively identified contact. These are appear as broken lines but share the color characteristics of friendly range rings.

Merge Range Symbols

This merges all range rings of a particular side to unclutter the UI. Be aware that it only displays the longest-ranged of the weapons, so that, for example, an SA-2 site will be drowned out by a much longer ranged SA-5 site nearby.

Sonbouy Visibility

This changes sonobouy visibility. For details, see “Map Display” under “Game Options” above.

Reference Point Visibility

This changes reference point visibility. For details, see “Map Display” under “Game Options” above.

Illumination Vectors

Many modern weapon systems use or require a fire-control radar to illuminate a target to guide weapons to them or improve their accuracy. This includes most modern deck guns and many surface-to-air missiles. This is displayed as a broken red line leading from shooter to target.

Using the drop-down menu, you can set this to be displayed for: the selected unit, all units or None by clicking on your choice in the drop down. It is set to None by default to lower the clutter on the display.

Targeting Vectors

Displays a broken green line from shooter to target and a small text date block below showing how target was allocated (manual or auto) and the expected time of weapon release.

Using the drop-down menu you can set this to be displayed for: the selected unit, all units or None by clicking on your choice in the drop down. It is set to None by default to lower the clutter on the display

Datablocks

Displays unit name, course and speed to the right of unit in white text in the display.

Using the drop-down menu you can set this to be displayed for: the selected unit, all units or None by clicking on your choice in the drop down. Selected unit is set to be the default value.

Datalinks

Displays any communication link between units.

Using the drop-down menu you can set this to be displayed for: the selected unit, all units or none by clicking on your choice in the drop down. It is set to "None" by default to lower the clutter on the display.

Contact Emissions

This displays the emissions of certain non-friendly contacts on the display. It can be toggled to all, selected, or off. It can also toggle only fire control radars.

Plotted Courses

This displays the courses of plotted units in a line, and can be toggled to selected, all, or off.

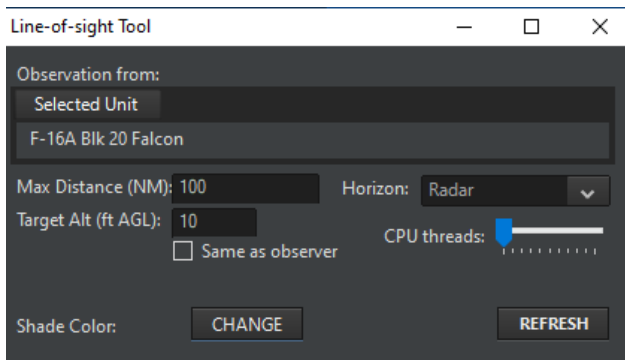
Mission Area/Course

This displays the mission area in the form of a highlighted box with the mission's name nearby and can be toggled to selected, all, or off.

Track Selected Unit.

This keeps the camera centered around a selected unit.

LOS Tool



This enables the player to view the “line of sight” of a selected unit. The line of sight will appear as a colored area around the unit, making it useful for determining what the unit in question can see or detect. The tool can be applied to either a friendly or detected enemy unit, meaning it can be used to either plot offensive (“hmm, this mountain is a nice radar blocker....”) or defensive (“Since my ground-based radar stations can’t see well over here, I should send my fighter patrols over to cover the gap”) strategies.



The line of sight tool in action, with lime green selected as the color. To the north is mostly barren desert where the tool can reach its max distance. To the south is a mountain range where it’s frequently blocked off.

There are several customizable options on the tool.

-Max distance. This sets the maximum range in nm the tool will extend to. It can be reined in either for the sake of performance or for showing a sensor with very limited range.

-Target altitude (m AGL). This shows the altitude at which the LOS target is determined. Higher flying units can obviously be seen from farther away, and the setting can be changed in light of the usual attack profile of the scenario’s threat units (ie, threat aircraft with terrain following suites like F-111s or Su-24s will go at lower altitudes than ones that don’t have them, like earlier and/or simpler aircraft). The “same as observer” box can be checked to assume that the target is at the same altitude as the selected unit.

-Horizon type. The “Radar/ESM” option, assuming an “electronic horizon” capable of reaching longer distances than visual sight, is the default. The shorter range “Visual/EO/Laser” option can be checked if need be, such as for a unit that uses those as its primary sighting system.

-Shade color. Clicking “change” opens up a palette of colors that can be selected as the one to display. Color can be changed to avoid conflicting with other map elements or for the sake of aesthetics/ease of sight.

-CPU Threads. This allows the number of CPU threads used in calculating LOS to be changed for performance reasons. (considered obsolete)

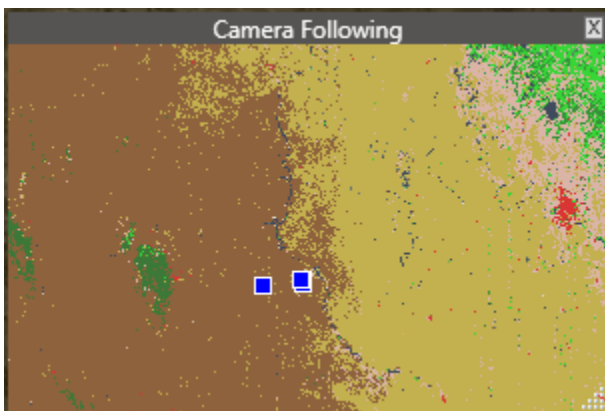
It is important to note that line of sight is different depending on the type of system selected. Many radar waves have the ability to travel 'over' the horizon to a limited extent resulting in a greater range beyond that of visual and IR sensors.

Minimaps

This toggles minimaps. There are three main types of minimap, all of which can be used at the same time.

- **Global.** This displays a map of the entire world. Only recommended for very large scenarios, as the unit contacts will otherwise appear to be incredibly close together.
- **Scenario.** This displays a map with borders established by the outermost units in the scenario. In other words, it displays a map of the theater of operations.
- **Camera-following.** Self-explanatory, this minimap zooms in and out with the camera, showing what the display currently shows.

On the minimap, units follow the same color code as on the main display- blue for friendly, green for neutral, orange for unfriendly, red for hostile, yellow for unknown. Minimaps appear in their own windows and can be adjusted to be as large or small as the player desires. The minimaps also preview the local terrain making it easier to identify the orientation and position of various units as it relates to the world. Minimaps can be adjusted for size by clicking and holding the bottom right corner of the map.



Terrain Type Legend

This toggles a legend that illustrates the types of terrain shown on the map when the "Land Cover" option is selected (see 6.2, View, and 3.1, The Globe Display). This is useful for seeing when a unit is in a concealing, shielding forest or exposed open land.

6.6 Quick Jump

Quick Jumps can be used to rapidly shift back to a unit or contact. To use quick jumps:

- Select a unit/contact** and zoom the camera to the desired level.
- Press CTRL+[NUM]** (the number can be 1, 2, 3, 4, etc...) and the camera altitude and location will be saved
- To jump to that unit/location, either press the initial number or use the "quick jump" top level menu and select from the list. The saved numeral is to the right of the unit in the menu for easy reference.
- Note that quick jumping doesn't select the unit/contact, instead it merely moves to its location.

6.7 Unit Orders Drop-Down Menu

This duplicates the right click/context menu.

6.8 Contacts Drop Down Menu

This menu is normally greyed out but becomes accessible once a contact is selected.

Drop Contact(s): This removes the selected contact(s) from the map. Note that they may be instantly re-acquired if the right sensors are there. This is for removing contacts like an undersea contact with an uncertainty area so large as to be useless, and which is probably just a whale anyway.

Mark (to posture): This allows the selected contact(s) to be manually marked as either friendly, neutral, unfriendly, or hostile.

Rename: This allows the contact to be renamed.

Mark Position: This creates a reference point at the contact's current location with the name being the name of the contact and the time the contact was seen at the position in question.

Filter Out All Contacts: This removes the individual visual filters on all non-side contacts.

Cancel Filter Out: This restores the individual visual filters on all non-side contacts.

6.9 Missions & Reference Points Drop-Down Menu

6.9.1 Mission Editor Menus

Mission Editor

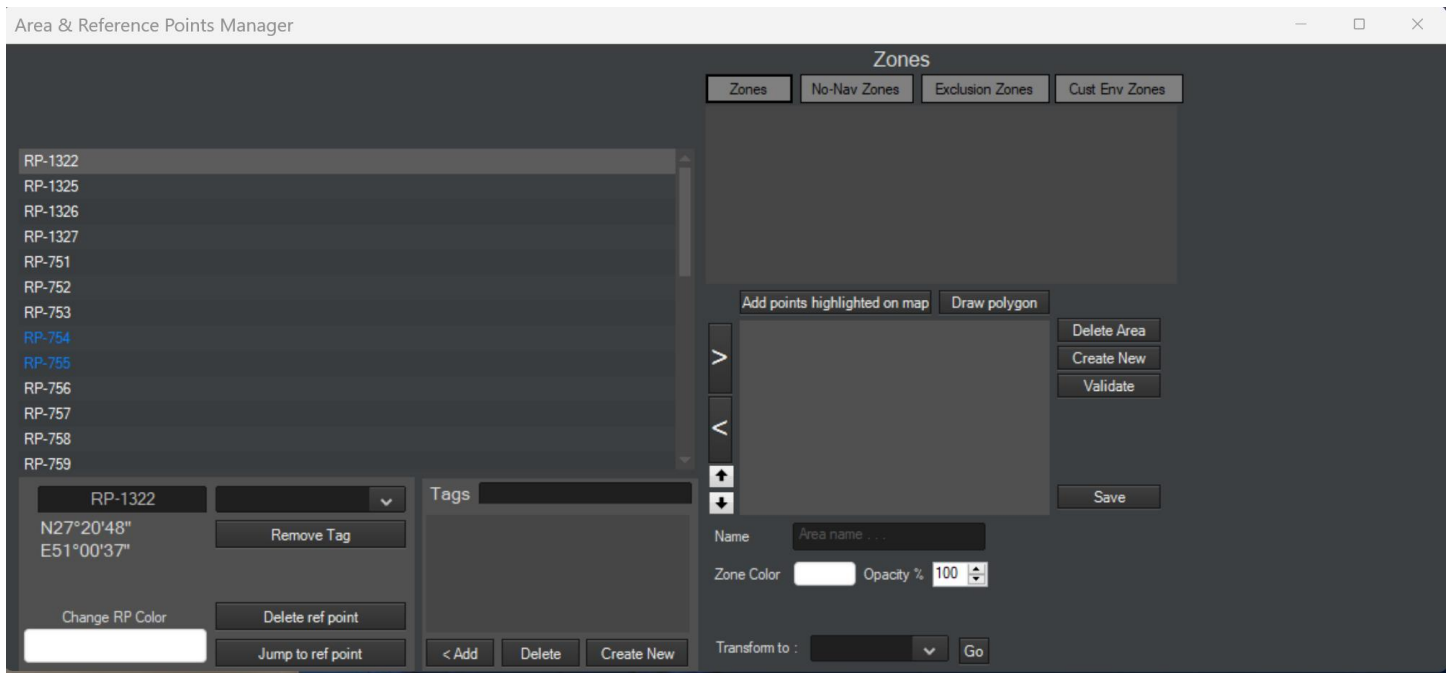
This opens up the Mission Editor window for quick access to existing missions. (See Section 7.2 for details on the mission editor.)

Add New Mission

Similar to clicking the “create new mission” button in the mission editor itself, this opens the “new mission” window.

6.9.2 Reference Point Menus

Area/Reference Points Manager: This allows you to manage reference points. No-navigation, exclusion, and custom environment zones that enable the player to change the weather in a specific area can be created.

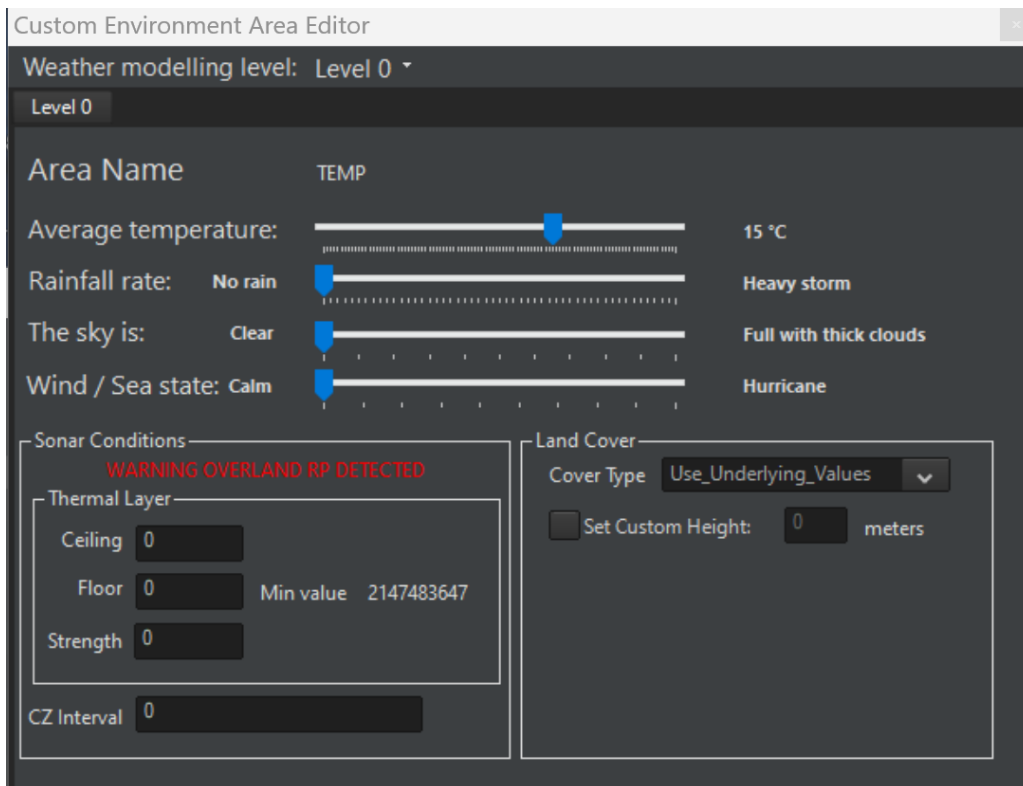


The reference points manager gives you the ability to apply metadata to individual reference points such as color and tags. These tags can then be filtered to create the appropriate selections for given missions. On the right side of the screen is a menu that allows you to create areas for reference points. Simply select the reference point you wish to add to an area on the left panel, press the target > button and then give the zone an appropriate name and press the save button. The bottom of the screen you have the **Transform to** dialog that lets you convert one type of zone to another.

From the box in the upper right, you can select existing zones to manage and edit:

- Zones – A zone is area defined by reference points and in this box, pressing the “Zones” button will show any zones defining an area that have not yet been assigned a specific function listed below.
- No Navigation Zone – A zone defining an area where specific unit types cannot enter. (see section 6.9.3)
- Exclusion Zones - Exclusion zones are designated areas that, if entered by an object not belonging to the player, such as a ship or planer, will change the player-sided posture to something the player has previously specified.
- Custom Environment Zone – A zone that allows creating specific the weather and/or terrain inside the defined zone. This zone then overrides the global weather or underlying terrain for the defined area only. (see notes on build 1307.6)

To make a custom environment zone, select “Cust Env Zones”, an appropriate number of reference points, select “Create New”, and then “Add Points Highlighted on Map”. Click “Edit” to change the weather and/or terrain inside the zone.



When finished, press “Save”.

Add Reference Point: After selecting this, the player may click anywhere on the map to place a reference point.

Define Reference Point Color: This allows the colors of reference points to be changed from the default white. It opens a standard color selection dialogue.

Delete Selected Reference Point(s): Pressing this removes all currently selected reference points. “Delete Waypoint” does the same for way points on a unit route.

Rename Selected Reference Point: This allows an individual reference point to be renamed. An error message will appear if multiple points are selected when this is clicked.

De-Select All Reference Points: This deselects all reference points on the map.

Define Area: Like the CTRL+Right Click dialogue, this allows for a box of reference points to be drawn on the map.

Make Selected Ref Points Relative (Fixed Bearing) to...: Selecting this and then selecting a unit/group will cause the reference points to become relative to, or move with the unit/group. One use of this is making an ASW patrol area for naval aircraft that moves with the group of ships it’s intended to guard.

Make Selected Ref Points Relative (Rotating Bearing) to...: Similar to the above, only this will cause the reference points to rotate with the units. So if the points are initially placed to the north of the subject and it does a 180 degree bearing change, fixed-bearing points will stay where they are while rotating bearing points will move with the turn until they are to the *south* of the subject.

Change Bearing Type: This makes relative reference points either fixed or rotating.

Remove Relativity from Selected Points: This returns relative reference points to normal. They will no longer “follow” any individual unit and will stay in a fixed location on the map unless manually moved.

Lock/Unlock Selected Reference Points: This ScenEdit feature allows reference points to be locked so that they cannot be moved in normal play.

6.9.3 No Navigation and Exclusion Zone Menus

No Navigation Zones: This function allows editors to define or edit areas where specific unit types cannot enter.

The Edit No Navigation Dialog has a list of zones at the top and specific parameters for each at the bottom. You can navigate between zones by selecting each from the above list.

To create a No-Navigation zone:

1. Make sure the side you would like to create a zone for is selected.
2. Create and select reference points on the map that will be used to mark the area that units will be excluded from
3. Select Create new from selected reference points from the No Navigation Zones drop down which will launch the Edit No Navigation Zones dialog.
4. Notice that the selected reference points have populated the Zone area list. You can adjust their positions by clicking and dragging them on the map. You can remove them by selecting them in the Zone area list and clicking the Remove Selected button.
5. You can assign unit types to the zone by checking off any or all of the Applies to check boxes.
6. You can name or rename the zone by filling in the description text field.
7. You can save the zone by clicking the Save button which will populate the zone list at the top.
8. To remove a zone, select it from the list at the top and click the remove button at the bottom of the dialog.

Exclusion Zones: Selecting this function allows the player to create an air, ship, submarine or land unit exclusion zones. Exclusion zones in COMMAND are designated areas that, if entered by an object not belonging to the player, such as a ship or planer, will change the player-sided posture to something the player has previously specified. The exclusion zone gets its name in part from the titular zone declared around the Falklands in 1982.

A good example of where this might be useful is if you wanted to create a “No Fly Zone” similar to what was over Iraq in the 1990s. You could designate an area and then make all air contacts flying into the zone be designated as unfriendly (to be intercepted but not fired up on) or hostile (to be immediately attacked) by creating an intercept mission designed to launch upon detection of a unfriendly or hostile target.

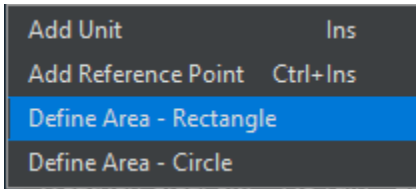
To create an exclusion zone:

1. Drop some reference points to mark an area. It is suggested you use the control right click function to define an area (drop 4 reference points). Please make sure they are activated by clicking them.
2. Click on Exclusion Zones from the Unit Orders drop down and select Create new from selected reference points or Edit Existing which will launch the Edit Exclusion Zones dialog.
3. Notice that your reference points now have a border drawn. You can click and drag reference points to change the zones shape.
4. You can designate the zone to apply to Aircraft, Ships, Submarines, land units or all by checking all off.

5. You can name the Zone by filling out the description.
6. You can specify a posture using the drop-down in the Mark Violators field.
7. You can designate and undesignated reference points using the Zone area fields, To add click and highlight one or more reference points on the map.
8. To remove reference points select them in the zone area field and click remove selected.
9. To save your exclusion zone click the save button.
10. To remove a zone, select the zone from the list at the top of the dialog and press the remove button.
11. To close, click the red X on the top right-hand side of the dialog.

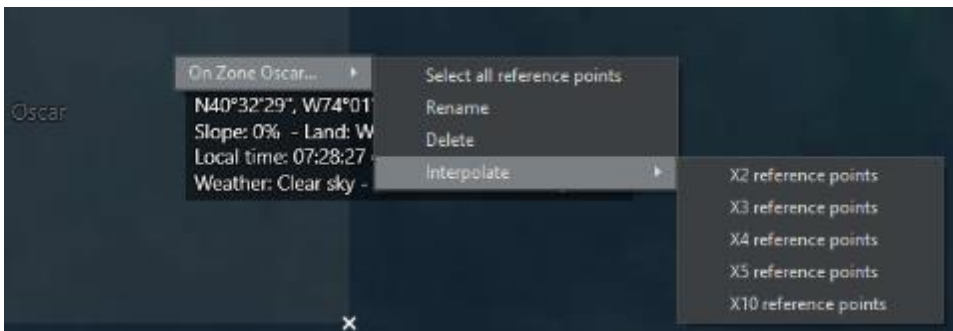
6.9.4 Quick Creation and Editing of Zones on the Map

Basic zones can be created on the world map by holding down the control key and right clicking anywhere on the map. A few options will appear:



After selecting one of the two areas, you can now left click and drag the shape of your zone onto the map. After releasing the mouse, a dialog will appear and ask if you would like to create a zone. If you select yes, a new dialog box will appear and ask you to name this zone.

Pre-existing zones on the world map can now be selected by shift-right clicking somewhere inside the zone. This will bring up the name of the zone in a dialog box and provide you with new options:



Select all reference points: This will quickly select all the reference points in an area. This will *deselect* all existing selected reference points.

Rename: Rename the zone

Delete: This will provide you with the option to delete the zone as well as the reference points for that zone if required.

Interpolate: This will create new reference points in the zone that give you the ability to manipulate the shape of the zone with greater granularity. Note that if you delete the reference point used to create a zone, the zone will retain that reference as a vertex for the shape.

When editing the points of a zone, if you hold down shift and left click, you can drag *the entire zone* as one group of reference points instead of moving them individually.

6.10 Help Menu

Coordinate Converter: Allows coordinates to be converted from one format to another.

Hotkeys: Displays a list of all hotkeys.

Tutorial+How To Videos: Displays an internet link to a series of tutorial videos on various topics.

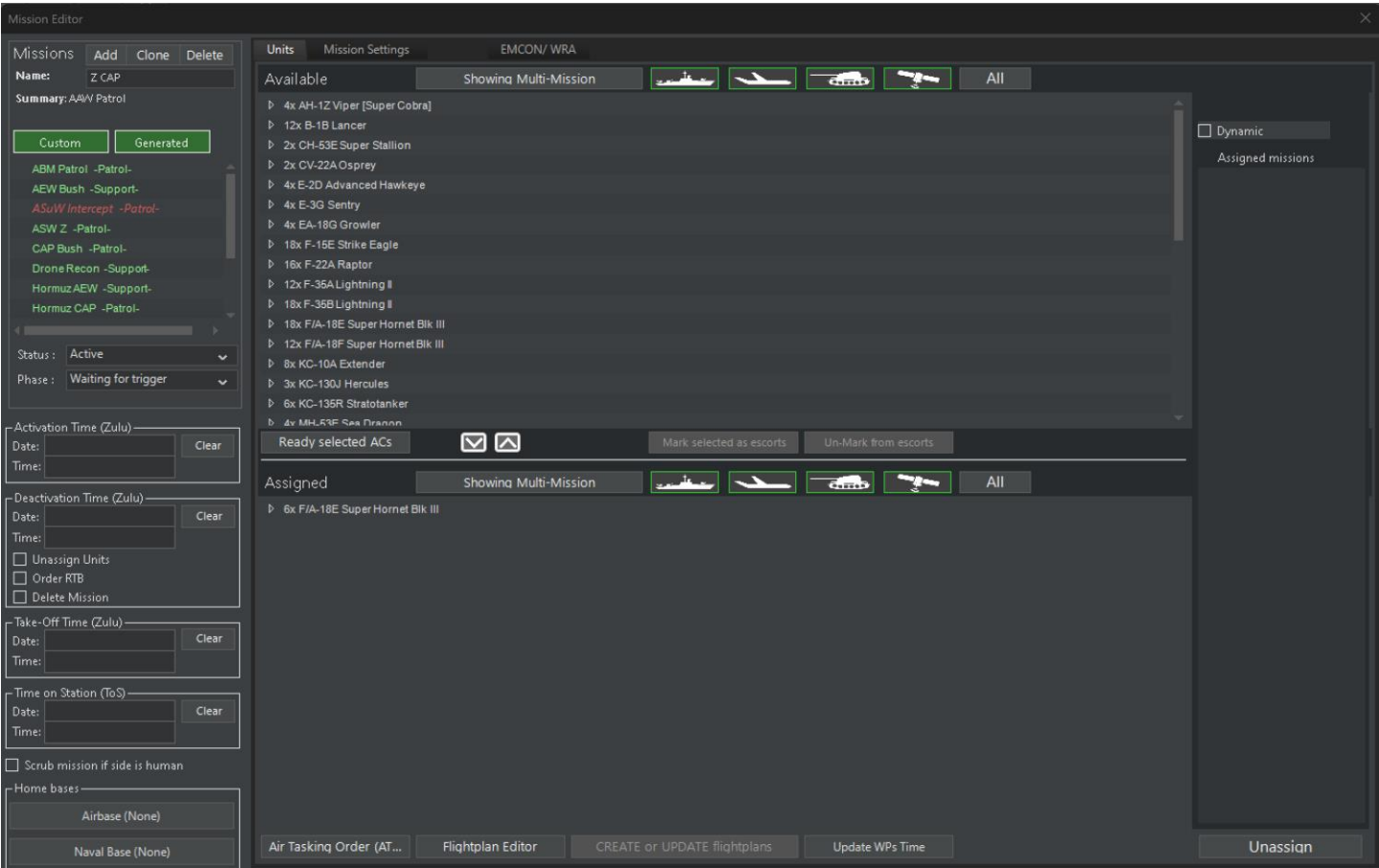
About COMMAND: Lists the credits and all modules currently licensed by the player.

7.0 Missions, Cargo Missions, Operations Planning, and Reference Points

Missions and reference points are the vital building blocks of COMMAND. Learning how to use them is essential for both playing and creating scenarios.

7.1 Mission Editor

The mission editor provides a list of all current missions and it also gives you the ability to create, edit and manage them. You can access it from the toolbar by pressing the mission editor button or by pressing F11 on your keyboard.

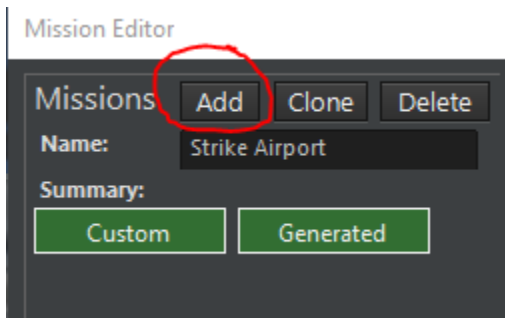


The mission editor is separated into three major tabs: units, mission settings and targets. Each page allows you customize the behavior of your missions as well as the relevant target assignment, formation settings and flight characteristics. When adding units to a mission, the type of loadout for each unit is displayed next to the unit to make it easier to identify which units you wish to use on a given mission.

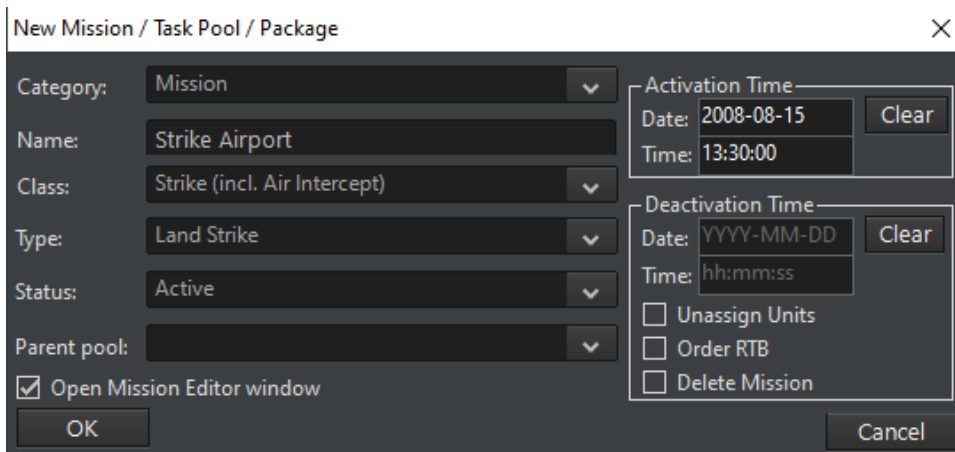
UH-60A Blackhawk (Troop Transport) (Airborne)

7.1.1 Adding a new Mission

To create a new mission, click on the Add button in the upper right corner of the dialog. Alternatively, you may press CTRL+F11 in the main game window to bring up the dialog. You also have the options to clone an existing mission by highlighting and pressing clone, or deleting an existing mission by highlighting it and pressing delete.



A box will appear allowing you to set the basics properties of your new mission.



The category of the mission refers to what type of mission you are planning to execute. Each category has a unique purpose:

Mission: This refers to your standard mission that various units can be assigned to. They are generally 'one-off' style operations that can be continuous but not easily repeated.

Task Pool : This is where you can define a group of units for the purposes of completing packages. Once a unit is added to the task pool, it can be assigned package missions.

Package: This type of mission refers to a single package in a larger mission and unlike regular missions, it requires units to be assigned to the task pool before it can be started.

Once you have selected the category of mission, you can select an appropriate name and class.

New Mission / Task Pool / Package

Category: Package

Name: Package 3018

Class: Strike (incl. Air Intercept)

Type: Strike (incl. Air Intercept)

Status: Support

Parent pool: Mine-clearing

☒ Open Mission

Take-Off Time

Date: YYYY-MM-DD Clear

Time: hh:mm:ss

Time On Target (ToT)

Date: YYYY-MM-DD Clear

Time: hh:mm:ss

OK Cancel

Mission classes are described later on in the documentation. Once the class is selected, then the type of mission is selected. Note that for strike type missions, the targets that are selected must match the class of mission for them to be completed properly.

After selecting the different types, a mission can be set of active meaning it will launch as soon as practical or inactive meaning that it will need to be activated later in order to start the mission. Task pools and packages do not have activation times and will have to be manually activated at a later time. Note that cargo missions that feature airdrops can be assigned into these groups so that you have more control over the various dropping units.

Activation Time

Date: 2008-08-15 Clear

Time: 14:00:00

Deactivation Time

Date: 2008-08-15 Clear

Time: 17:30:00

☒ Unassign Units

☒ Order RTB

☐ Delete Mission

When setting activation and deactivation time it is important to note that the times are listed in Zulu time and not local. Below the activation time window are options when the mission ends to:

Unassign Units – Remove all the units from this mission when it is over and put them back in the available aircraft pool after the mission is deactivated.

Order RTB – Order all units on this mission to return to base after the mission has deactivated

Delete Mission – Remove the mission from the mission list after it is deactivated.

Once the mission is created, the next steps are typically to set up appropriate times, doctrine and bases if required. Typically, time on target and takeoff times should be set after all the appropriate aircraft are assigned and other mission settings are developed. If multiple types of units are given the same time on target (such as ships with cruise missiles and aircraft with bombs), the units will reserve the attack so that they will not fire until they can strike the target as close to the same time as possible.

Missions
Add
Clone
Delete

Name:
Strike Airport

Summary:
Land Strike

Custom
Generated

(0) Strike Airport

Status:
Inactive

Phase:
Waiting for trigger

Doctrine, EMCON, WRA for:
Mission
Escort

Activation Time (Zulu)
Date:
2008-08-15
Clear
Time:
14:00:00

Deactivation Time (Zulu)
Date:
2008-08-15
Clear
Time:
17:30:00
☒ Unassign Units
☒ Order RTB
☐ Delete Mission

Take-Off Time (Zulu)
Date:
Clear
Time:

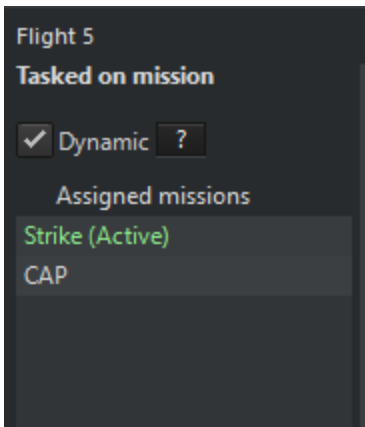
Time On Target (ToT) (Zulu)
Date:
Clear
Time:

☐ Scrub mission if side is human

Home bases:
Airbase (None)
Naval base (None)

The next step in building a mission is selecting the appropriate units from the available list and then pressing the down arrow. You can filter the visible units by pressing the appropriate icon at the top of the available units page. If you wish for a unit to be available for multiple missions, you can select that unit and then press the Dynamic button in the top right of the page. Once this pressed, the unit will remain available for tasking when 'Showing Multi-Mission' is selected regardless if it is engaged in a current mission. This is a powerful feature and requires careful timing to ensure the unit will be available properly loaded and configured at the moment it is required in a later mission.

Multi-mission units require you to select the unit and then click the checkbox next to the Dynamic[?] button. Once a unit is declared multi-mission, a list of the missions it is assigned to becomes available underneath this checkbox on the right side of the screen. If a unit is used in a multi-mission situation, the other queued missions will not be triggered until the previous ones have been marked as satisfied.



Note that for a multi-mission unit to be able to go onto new missions, the previous missions must be completed or satisfied.

At the middle of the units page there are options to filter the units on the mission as well as:

Ready Selected ACs: Selecting a checkbox next to an aircraft in the list and pressing this button calls the ready aircraft dialog. This allows you to make quick changes to aircraft loadouts.

Mark aircraft as escorts: Selecting checkboxes next to aircraft allows you to set them to be air to air or SEAD escorts within a strike group. When you do you will notice a [Escort] marker in their descriptions. Escorts are generally more useful for gun-only or early missile age “close escort” (like the classical WWII close escort) than for more recent scenarios with longer-range AAMs. For that period, a separate patrol mission is recommended.

Un-mark from escorts: Selecting checkboxes next to aircraft set to escort sets them to conduct normal mission behavior and not escort behavior.

The fields, buttons and dialogs at the bottom of the dialog allow you to change mission behaviors or parameters. Some will vary according to the individual mission in question, and are detailed in their sections.

Once the appropriate units are assigned, the mission settings tab has several options related to the mission. This page changes depending on the type of mission selected and each page will be described under their respective mission type.

Units
Mission Settings
EMCON/ WRA

Patrol

Try to keep 0 units per-class on-station (0 to ignore)
☒ 1/3rd rule (evaluated by a/c type and loadout)
☐ Investigate unknown contacts outside the patrol area
☐ Investigate unknown contacts within weapon range
☒ Active emissions only inside patrol / prosecution area (requires active EMCON)
Movement style: Random within area

Patrol Area

Area / RP manager

RP-751
RP-782

Pick area
Create area

Add points highlighted on map
Highlight + center
Delete
Validate Area

Aircraft settings
Non-AC settings

Flight size: 2 ☒ Enforce flight size (by base, AC type and loadout)
Minimum number of a/c required to trigger mission: 1x flight
Tankers (AAR): Inherited, Allow, but not tankers refuelling tankers
Cruise formation:
Attack formation:
Number of a/c that investigate unknown contacts: 1x flight
Number of a/c that engage hostile contacts: All flights
Wingmen can investigate/engage separate contacts within 5 nm
Time-of-day:
Weather:
Disable QRA if number of aircraft drops below:

Configure

Speed / Alt
Flight Plan

Aircraft Speed and Altitude

Transit throttle: Cruise
Station throttle: Loiter
Attack throttle: Unspecified
Transit altitude: High (36000 ft)
Station altitude: High (36000 ft)
Attack altitude: High (36000 ft)
Attack dist.: nm

A common feature of all mission settings is setting the minimum number of units to launch the mission as well as the option to add the flights to the air tasking order. The final tab of the mission editor page is the targets page. This page will only show if the mission requires targets.

Use tankers assigned to specific missions: This will open up a list of the missions with tanker aircraft currently assigned to them, with one being selectable. This can be used to prevent large aircraft (ie, heavy bombers) from quickly exhausting small tankers.

Minimum Number Of Tankers: This will prevent the mission from launching if the required number of tankers can't be met. It can be further set to "minimum number of tankers airborne" or "minimum number of tankers on station."

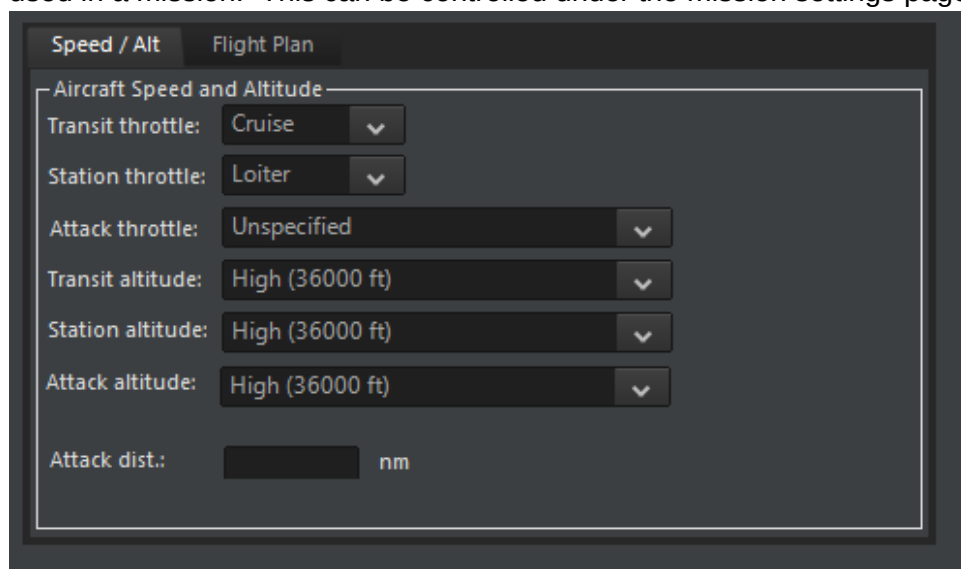
Launch Mission Without Tankers in Place: This is an extremely high-risk option to launch combat aircraft without tankers currently airborne. The aircraft will refuel, but only for tankers ahead of them and not behind them. This is playing with fire, so if they crash, don't say we didn't warn you.

Maximum Number Of Receivers In Queue Per Tanker: This allows the player to set limits on the number of aircraft "waiting in line" for each tanker to prevent pileups.

Receivers Start Looking For Tankers When Down To _____ : This allows the player to change the fuel threshold when aircraft start heading for tankers.

Airborne Receivers Can Book Tankers Within....: This allows the player to only allow the aircraft to look for tankers within a certain radius. This can be useful to avoid aircraft going to faraway tankers supporting a completely different mission.

Many types of missions give the ability to select a custom speed, altitude and depth for the types of units being used in a mission. This can be controlled under the mission settings page



Note that adjusting throttle may impact the maximum range of the mission and any flight plans that were created along with the mission. The Attack dist.: option allows you to define the desired distance to attack a unit. Keep in mind that needs to agree with the WRA of the unit as well as weapons' limitations.

7.2 Mission Definitions and Details

AI-controlled missions are tasks that platforms or groups of platforms can be assigned. The various mission types can have sub-types and their own Doctrine and EMCON. Missions are either Area Oriented or Task Oriented. Area Oriented missions such as Patrol Missions are defined by Reference Points, Task oriented missions such as Strike are defined by target units, Reference Points (area)

Selecting a zone/area based mission in the mission editor will show an outline of the mission area on the map.



Missions that are area based may be defined by selecting the appropriate reference points on the map, or if preferred, the pick area button may be selected in the missions setting page. Once this is selected, a menu appears with areas that can be picked for the mission. Note that if any existing reference points are selected, this may cause the areas to 'join' each other.

7.2.1 Ferry Mission

A Ferry Mission transfers an air unit from one location to another. The destination for a Ferry Mission is always a Unit, Facility, or Base that can house the unit to be ferried. This can include friendly and neutral air facilities. Ferry missions are highly useful for simulating civilian air traffic or "escape" missions like the Iraqi Air Force's flight to Iran in 1991.

To create a ferry mission:

1. Select the destination unit or facility
2. Add a new mission and when the dialog launches add a mission name and select the ferry mission type. Click the ok button which will launch the mission editor.
3. Assign the desired units to the mission
4. Set the Mission Doctrine/RoE/EMCON settings by pressing the button and adjust the desired altitude/speed under the "mission altitude" window (a routine commercial flight is going to be flying higher and slower than a lower altitude escape).
5. Set the Ferry Mission specific settings "Ferry Behavior":
 - a. One-Way: The mission will execute one time and will automatically scrub once complete.
 - b. Cycle: The mission will cycle between the start and destination points after the normal turnaround time for that unit at each end.
 - c. Random: The mission will randomly trigger a cycle.

Ferry Mission Settings:

Ferry Behavior: This allows the mission designer to choose between round-trip cycles or ordinary one-way trips. If "One way" is selected, the unit will stay at the destination base after it arrives there. If "Cycle" is

selected, the unit will arrive at the destination base, ready, leave the destination base once its ready time has elapsed, move back to its original base, then ready, then move back to the destination base.

“Random” randomizes what type of ferry behavior the unit will use.

7.2.2 Support Mission

Support missions allow players to assign specific units to follow waypoint-driven paths. They are most useful for AEW (airborne early warning), Refueling (air to air refueling) and reconnaissance missions.

To create a support mission:

1. Create and Select the desired Reference Points on the map.
2. Create a new mission, select the support mission category and type in a name for the mission. Click the OK button the launch the mission editor.
3. A line will appear on the map showing the support mission course (loop) path. The loop path may be adjusted by moving or editing the reference points.
4. Assign the desired unit(s) to the mission.
5. Set the Mission Doctrine/RoE/EMCON settings by pressing the button and set the desired throttle/altitude settings.
6. Set the Support Mission specific settings:
 - 1/3 RULE: If checked, will keep 1/3 of the available aircraft airborne if possible.
 - ONE-TIME ONLY: The mission will execute one time only and will be deleted and assigned units freed after completion.
 - Navigation Type:
 - a. Continuous Loop: The A/C will loop around the mission’s assigned reference points until Bingo Fuel.
 - b. Single Loop: The A/C will make one circuit around the mission’s assigned reference points and return to its base.
 - Transit Throttle: Throttle setting for the aircraft while transiting to the first mission reference point. Settings are Loiter/Creep, Cruise, Full, Flank/AB.
 - Loop Throttle: Throttle setting for the aircraft while executing the support loop. Settings are Loiter/Creep, Cruise, Full, Flank/AB.
 - Mission Reference Points. Reference points may be added or deleted to/from the mission. To add make sure the reference points are selected on the map.
7. Verify on the map the Support Mission path shown with a highlighted line.
8. For tankers alone, checking the “tankers return to base after one refueling cycle, when queue is empty” means that tankers will refuel one “queue-full” of aircraft and then return to base themselves. This can be used to keep them out of danger or avoid using too much fuel.
9. Likewise, for tankers, “tankers can refuel x numbers of receivers, rounded up to the nearest flight” (so for three receivers set at maximum, it will refuel two flights of two aircraft each) can be used to also restrain them.

Support Mission Settings:

Try To Keep (___) Units Per Class On Station: This sets how many units the mission will try to keep on the support track. Setting this to 0 means it will be ignored.

1/3 Rule: If checked, only a third of the aircraft currently assigned to the mission will launch and stay in on the support track at any one time. In the event of a conflict with the “Keep ____ Units On Station” setting, the larger number prevails.

One Time Only: If checked, the unit will stay on the support track as normal, but then return to base and stay there once it reaches its limits. It is not the same as a “single loop”

Navigation Type: “Continuous Loop” will have the unit(s) continually moving around the support track as long as possible. “Single Loop” will have the unit(s) move around the support track once and then return to base.

A tanker, AEW, or electronic warfare aircraft intended to remain on station as long as possible should receive a continuous loop mission. A recon aircraft making one pass should receive a single loop mission.

7.2.3 Patrol Mission

A patrol mission is an area mission defined by reference points. Any mobile unit may be assigned a patrol mission. If a facility or group is assigned this mission, all of its A/C will be assigned this mission. Aircraft will stay on patrol as long as their fuel is sufficient and their weapons fit the patrol settings. Submarines assigned to patrol missions will, by default, not use their active radars when at periscope depth or on the surface even if the inherited side EMCON is to emit if possible. The EMCON can be manually changed to have them do so if necessary (ie, when using submarines with poor sonar against weak opposition)

Patrol Mission Types:

- AAW PATROL: Units assigned to this mission will actively search for and investigate/identify airborne contacts.
- ASW PATROL: Units assigned to this mission will actively search for and investigate/identify sub-surface contacts.
- ASuW PATROL (NAVAL): Units assigned to this mission will actively search and investigate/identify waterborne contacts.
- ASuW PATROL (GROUND): Units assigned to this mission will actively search and investigate/identify contacts on land.
- ASuW PATROL (MIXED): Units assigned to this mission will actively search and investigate/identify any non-submerged contact.
- SEAD PATROL: Units assigned to this mission will actively search and engage targets emitting radar.
- SEA CONTROL PATROL: Unit assigned to this mission with actively search and investigate/identify naval surface and undersea contacts.

To create a patrol mission:

1. Create/select at least three reference points on the map to define the patrol area.
2. Create a mission, select the patrol mission class and types from dropdowns, type in a name and click the OK button which will open the mission editor.
3. A shaded box is shown on the map depicting the patrol area. A folded box or bow-tie shape is the result of the Reference Points not being placed sequentially around the perimeter of the desired area. It is best to use the right-control-click Define Area function as it will drop four reference points in the correct order.
4. If desired, create a prosecution area. Create/select a second set of reference points, press the “prosecution area” tab, and add/remove them as one would for a normal patrol area. In order for the prosecution area to work, “Investigate Contacts Outside Patrol Area” must be checked.
5. Assign appropriate units to the mission.

6. Set the Mission Doctrine/RoE/EMCON settings by pressing the button and set the desired throttle/altitude settings.
7. Set 1/3 RULE: If checked, will keep 1/3 of the available aircraft airborne if possible.
8. Set "Investigate Contacts Outside Patrol Area" If checked, the units assigned to the mission will leave the patrol area to investigate/identify nearby contacts. If unchecked they will restrict their activities only to their defined area. If checked with the mission containing a prosecution area, the units will only investigate contacts inside said prosecution area.
9. Set "Active Emissions Only Inside Patrol Area" If checked, the units will transit to the patrol area with radars off to avoid giving away the location of their parent unit.

Patrol Mission Settings:

Try To Keep (___) Units Per Class On Station: This sets how many units the mission will try to keep in the patrol area. Setting this to 0 means it will be ignored.

1/3 Rule: If checked, only a third of the aircraft currently assigned to the mission will launch and stay in the patrol area at any one time. This is so that continuous coverage can happen, with some aircraft flying and others down for readying. In the event of a conflict with the "Keep ___ Units On Station" setting, the larger number prevails.

Flight/Group Size: Determines how many units are in each flight/group. Flights/groups will move as one "unit" on one path.

Tankers/AAR: Sets tanker/AAR settings, from "don't allow" to "allow, including tankers refueling each other".

Number of Aircraft/Units That Investigate Unknown Contacts: This allows the player to choose how many aircraft/boats/groups will abandon their original patrol pattern to pursue and investigate an unknown contact. It can range from none at all to every single deployed unit/flight. Limiting the number of units that do this prevents "pileups" where say, twenty F-15s enforcing a large no-fly zone all rush to investigate an unknown obvious civilian aircraft.

Number of Aircraft/Units That Engage Hostile Contacts: This is like the above, but applied only to engaging targets confirmed/marked as outright hostile. The default is all flights because of a perceived need to "better be safe than sorry", but it can be adjusted as needed.

Wingmen/Group Members Can Investigate/Engage Separate Contacts within: This sets the distance at which components of a group will split off and interact with separate contacts. The default is a short 5 nautical miles, but it can be raised or lowered as needed.

Movement Style: This sets the basic nature of the patrol. "Random Within Area" has randomly generated paths throughout the assigned "box" of reference points and was the default patrol system used in previous versions of COMMAND. "Repeated Loop" treats the mission reference points as parts of a "track", much like support missions, and the map display of a patrol mission set to "repeated loop" resembles that of a support mission.

Unlike a support mission, a "repeated loop" patrol maintains the more aggressive "investigate and engage if need be" (as doctrine and patrol settings permit, of course) logic of proper patrol missions. Thus an exact,

repeatable pattern can be generated using that method. Using a prosecution area for a “repeated loop” patrol is recommended.

7.2.4 Strike Mission

This mission type also includes Air Intercepts. Strike missions are attack missions with specific or categorized targets and defined attackers. Aircraft at airbases and naval units at ports/motherships will sortie on a strike mission and then return to base when the target is destroyed, unlike in patrol missions. Thus, if a continuous presence is desired, patrol missions may be more valuable than strike missions. On the other side of the coin, if because of distance, risk, or quick-turnaround times, a continuous presence is impossible, a strike mission may be preferable to its patrol counterpart.

Strike Mission Types:

- **AIR INTERCEPT:** Units assigned to this mission will attempt to intercept and engage the selected Air Contacts.
- **LAND STRIKE:** Units assigned to this mission will transit to within the selected weapon’s range and engage the selected land targets.
- **NAVAL ASuW STRIKE.** Units assigned to this mission will transit to within the selected weapon’s range and engage the selected surface contacts.
- **ASW STRIKE:** Units assigned to this mission will transit to within the selected weapon’s range and engage the selected submerged contacts.

To create a strike mission against a categorized target:

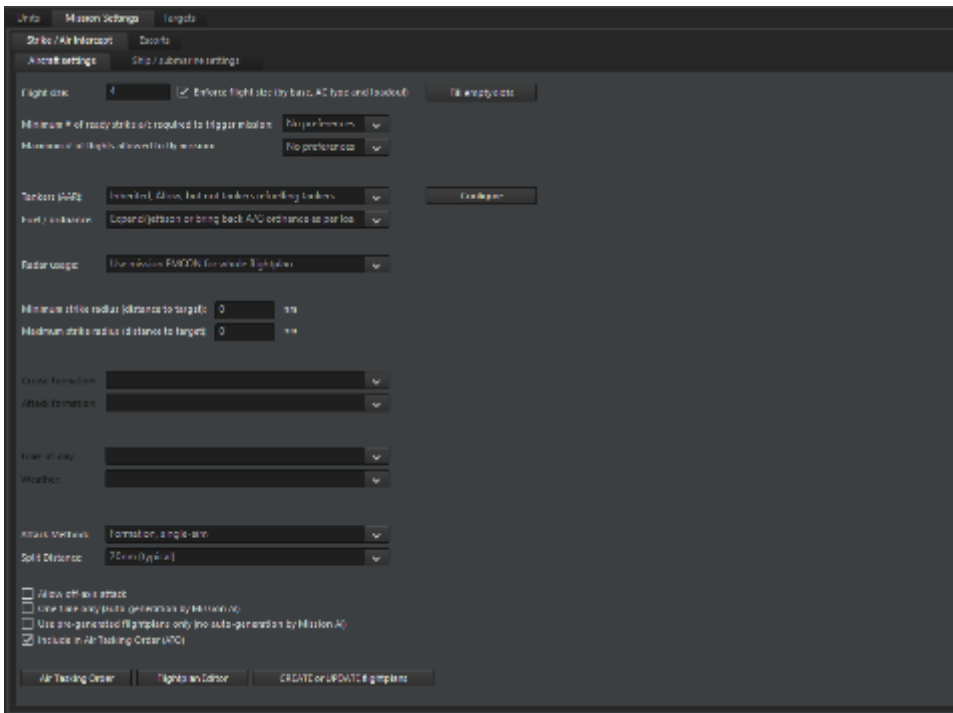
1. Add a new mission and choose the strike mission class.
2. Choose a mission type (air intercept, land strike, naval AsuW strike, ASW Strike) which defines the category of targets your unit will hunt for.
3. Click OK, which will launch the mission editor.
4. Assign appropriate units to the mission and set the desired unit parameters, including escorts if possible.
5. Set the Mission Doctrine/RoE/EMCON settings.
6. Define trigger options

Keep in mind that this mission launches against targets on a first-detect basis. It is most useful for ASW or intercept strikes.

To create a strike mission against a specific target or set of targets:

1. Select or group select a set of targets on the map.
2. Add a new mission and choose the strike mission class and category you would like to use. Click the OK button when complete which launches the mission editor.
3. Assign the appropriate unit to the mission. You can look in the platform display to select the best platform and aircraft loadout to use.
4. Select an appropriate mission trigger.
5. Set the Mission Doctrine/RoE/EMCON settings.
6. Click Add Units to populate the Target List. If you’d like to remove any units, select the entry in the list and click the Remove Selected button.

Additionally, the mission settings page allows for further customization of the mission and the methods of engagement.



The options include:

Flight Size – This dictates how large of flights will be for the strikers or for the escorts. If the Enforce flight size box is checked, then a group of units smaller than the size will not launch on the mission.

Minimum # of ready strike a/c required to trigger mission – This option sets the minimum number of units to be ready for deployment before the mission will start.

Maximum # of flights allowed to fly mission – This sets the total number of groups to be launched on the mission. This is useful for continuous missions where a suitable reserve number of strikers or interceptors can be deployed after previous groups have disengaged.

Tankers (AAR) - This is described in 7.1.1.

Fuel / Ordnance – This option determines how a unit will handle situations where it is unable to strike its target. Since the maximum range of a unit is dictated by its payload, opting to retain ordnance at long ranges may result in an aircraft unable to return to base without jettisoning its load.

Radar usage – This allows the radar usage to be dictated based on the scenario. The radar usage can be set to use existing EMCON, activate radar at IP or activate radar at attack leg.

Minimum and Maximum strike radius – Using this allows a mission designer to tweak the ranges that weapons are deployed. If left at zero, the weapons will be utilized as soon as they come into the appropriate to do so. This is a good way to force certain weapon carriers to engaging within a set range band.

Attack Method – This sets what tactical maneuver should be used to strike the target. The options range from large formations striking at one target, to independently targeting after splitting into smaller units before striking.

Split Distance – If an attack method was selected that involves breaking a formation up to strike at multiple angles, the split distance dictates how far from the target the formation should split up before continuing onto the IP.

Allow off-axis attack – This option tells the mission editor to try to offset the arrival positions of multiple groups to not strike from the same direction. It can be used in conjunction or independently of the attack method options.

Use pre-generated flightplans only – Checking this tells the mission editor that the user intends to manually create and update the flight plans for the mission and not to generate at the time the mission is triggered.

Include in Air Tasking Order (ATO) - This adds the aircraft used in this mission to the ATO.

Air Tasking Order – This page features all the missions and packages currently assigned in the mission editor and is detailed in a different section of the manual.

Flightplan Editor – Pressing this button will bring up the flightplan editor. Before pressing this option, it is suggested to assign the required flights first, then set the appropriate takeoff or time on target.

Flightplan Editor for flight Nail76
✕

Selected Flight

Package: StrikeOne
Flight: Nail76

Create Flightplan

Create Full Flightplan

Delete Flightplan

Copy Flightplan

Callsign: Nail76
Type: Flightplan
Status: Planning
Task: Strike (Land)

Fill Empty a/c Slots

Change a/c type and loadout

Change take-off time

Change objective time

Change take-off loca...

Change landing locat...

Change diversion loc...

Aircraft: F/A-18F Super Hornet
Loadout: GBU-24E/B GPS/LGB [BLU-109A/B], ATFLIR [FLIR]

Zulu take-off time: 008-08-15, 13:27:48
Zulu objective time: 08-08-15, 14:15:00
Take-off location: CVN 65 Enterprise
Landing location: Not set
Diversion location:

ADNA #5
IOWA #4
IOWA #3
IOWA #2
IOWA #1
ADNA #12

#	Type	Zulu Time	Local Time	Speed	Altitude	Leg Distance	Total Distance
1	Take-Off	13:27:48	05:27:48 (Day)	Loiter (350 kt)	2000 ft ASL	-	-
2	Hold Start	13:30:12	05:30:12 (Day)	Loiter (350 kt)	2000 ft ASL	12 nm	-
3	Hold End	13:39:48	05:39:48 (Day)	Cruise (480 kt)	36000 ft ASL	-	-
4	Turning Point (Ingress)	14:06:28	06:06:28 (Day)	Cruise (480 kt)	12000 ft ASL	178 nm	-
5	Weapon Launch	14:13:58	06:13:58 (Day)	Cruise (480 kt)	12000 ft ASL	50 nm	-
-	Weapon Target	14:15:00	06:15:00 (Day)	-	-	5 nm	-
6	Turning Point (Egress)	14:22:02	06:22:02 (Day)	Cruise (480 kt)	36000 ft ASL	59 nm	-
7	Landing Marshal Point	14:46:21	06:46:21 (Day)	Cruise (480 kt)	3000 ft ASL	194 nm	-
8	Land	14:47:48	06:47:48 (Day)	Loiter (350 kt)	Minimum	12 nm	-

Insert Waypoint
Delete Waypoint
Edit Speed / Altitude
Edit Time
Clear Time

Edit Doctrine / EMCON / WRA
Edit Air-to-Air Refuelling Settings
Edit Sensor Usage

Targeteering & Weaponeering

The flightplan editor allows very careful manipulation of flight plans including speeds, positions, altitudes and times. When using the flightplan editor, it is suggested to press create or update flight plans in the main Mission Settings page first to pre-generate the default flight plans. The features of this page are below:

Package - This is the currently selected package or mission that is being edited

Flight – This is the currently selected flight. If create or update flightplans has not been selected, this may be empty.

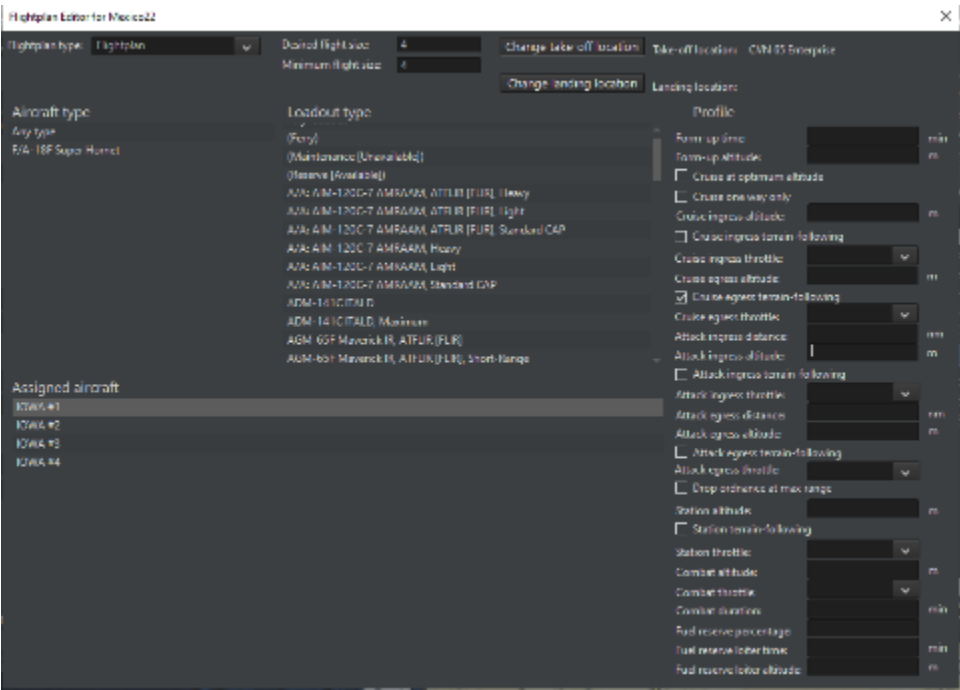
Callsign – This is the callsign of the given flight. Changing this will also change the name of the current flight.

Task – This is the currently assigned task of this flight.

Note: The flight plan generator is a very intricate mechanism. Newer players should keep mission building and flight plans simple until they become more experienced in complex operations.

Along the right side of the screen are several options related to flight plans and major time control for a given unit. This is especially useful for tweaking missions so that different groups of units are able to have different characteristics depending on the tactical needs of the operation.

The change a/c type and loadout page enables a greater degree of customization for a given flight and the weapons that are carried by it.:

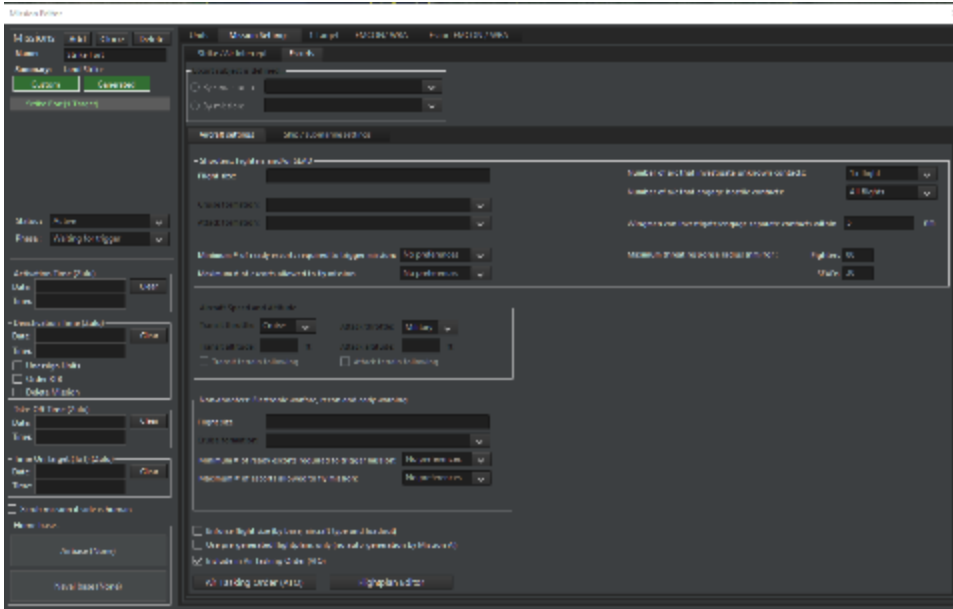


The change take-off and objective times allow for splitting off the various timings for the given flight independent of the main mission takeoff and target times. Selecting a unit on the main map that can be used as a landing area allows you to change the take-off, landing and diversion locations respectively. If the takeoff location is not the units current location, the aircraft will be removed from the flight plan and the option for an empty slot will be placed.

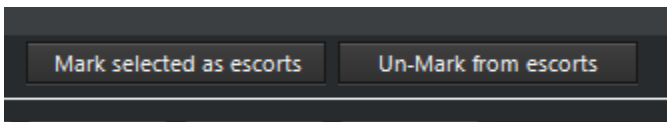
Below the change buttons is a table that provides critical information about each unit in a flight along with several adjustable features that can be set by selecting the appropriate waypoint and then clicking the buttons located below. Note that if you lock a specific speed or time, you can create a situation where an aircraft will run out of range or time in an effort to match the desired situation. Other elements of a waypoint can be edited by clicking on the down arrow next to their respective option.

The buttons along the base of the flight plan editor allow for the inserting of waypoints, deletion of waypoints, editing the various properties and times of the waypoints and items such as doctrine and sensor usage.

Setting up escorts is done in much the same way as the regular strikers, the differences being in how they respond to targets that could threaten their charges.



Aircraft can be marked as escorts using the units page by selecting them and clicking the appropriate button to add or remove them.



The unique features of the escort page are:

Number of a/c that investigate unknown contacts – This defines how many aircraft will check out an unknown target detected within their threat radius. This is a useful item to check to limit how many flights will investigate each unknown that comes along during the mission.

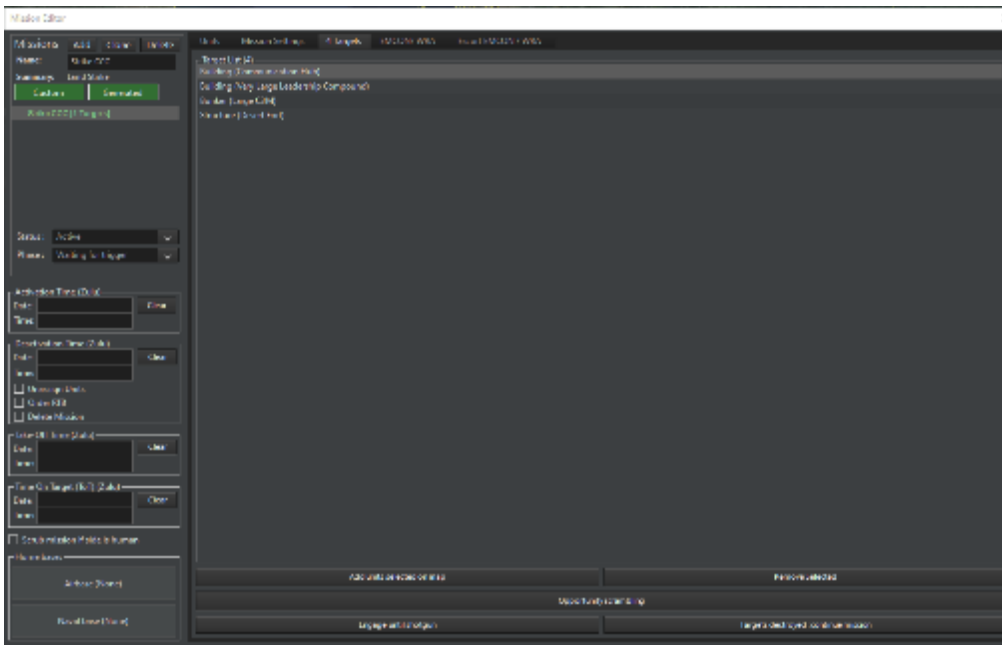
Number of a/c that engage hostile contacts – This sets how many flights can engage each hostile contact that is in the threat radius.

Wingmen can investigate/engage separate contacts within – This variable allows you to define how far apart wingmen escorts can operate when checking out contacts.

Maximum threat response radius – This defines how ‘far away’ escorts can go to engage a threat to the mission.

The escort page also features a separate page for ships and submarines, shooters and SEAD and non-shooters such as EW aircraft allowing you to have granular control of the package sizes involved. At the base of the page, you will find the air tasking order option as well as the use of pre-generated flight plans, these do not affect the escorts since they will automatically join up with the strikers on a mission.

The targets page allows you to control what targets are selected for a given strike mission.



All targets that have been added are located in the middle of the screen. If more need to be added, it is simply a matter of selecting them on the world map and then clicking the 'Add units selected on map' button at the base of the screen. Removing targets is a matter of clicking the target on the list and then hitting remove selected. If you need to select multiple, you can hold shift to select them by list or control to select them individually. At the base of the targets page are additional options:

Opportunity Scrambling – This sets whether or not the strikers and escorts will take shots at other targets along the way to the mission area. If this is set to off, the escorts still work normally.

Engage until shotgun – This allows you to quickly set the engagement parameter between shotgun (a single engagement) and winchester (continue engaging until the appropriate weapon is depleted).

Targets destroyed – This allows you to define whether to continue the mission after the targets have been destroyed (useful if the mission has targets added to it in situ) or to return to base immediately after the destruction.

7.2.5 Mining Mission

All mining missions are defined, in terms of area, by reference points. Units assigned to this mission will place mines randomly within the defined area, must be capable of deploying mines, and must have mines available.

To Create a Mining Mission:

1. Create and select the reference points designating the area to be mined. Try and find a good balance between "too small" and "too thinly-spread".
2. Add a new mission and select the Mining Mission type.
3. Add suitable units to the mission. The only effective units for this mission are certain surface ships, submarines, and aircraft that are specially equipped to dispense mines. The mine layer must also have mines available.

4. Set the Mission Doctrine/ RoE/ EMCON by pressing the button. Set the throttle and altitude settings if need be, but be sure to stay within the release parameters of the mines in question.
5. Set 1/3 RULE: If checked, will keep 1/3 of the available aircraft airborne if possible. Has no effect if aircraft are not assigned to the mission.
6. Set the "Arming Delay." This sets the amount of time before the mines will arm after being deployed. The default setting is 2 hours. This time allows the unit(s) laying the mines to safely exit the area.
7. Verify the proper reference points are activated and add by pressing the Add Points Currently Highlighted on Map button, which populates the Edit Mining Area reference point list. You can remove any points by selecting them in this list and pressing the Remove Selected button.
8. Ensure that the mines are satisfactorily dropped-they can fall in strange patterns.

7.2.6 Mine-Clearing Mission

All mine-clearing missions are area missions defined by reference points. This mission type is also used to locate mines. Units assigned to this mission will patrol their mission area hunting for mines. If mine(s) are located, the unit(s) will maneuver in such a way to place them in their sweep path. Only units equipped to sweep mines can do so; note that they still risk damage during the sweep operation. Units equipped with mine countermeasures equipment will have a "MCM" button on their information panel. MCM-capable helicopter units have a variety of specific load-outs for detection and sweeping, some specific to a type of mine. Not all mines are sweepable.

Creating a Mine Clearing Mission:

1. Create and select the reference points designating the area to be cleared of mines.
2. Create a new mission, select the Mine Clearing Mission category and type in a mission name. Click OK and the mission editor will launch.
3. Add suitable units to the mission. The only effective units for this mission are certain surface ships, submarines, and aircraft that are specially equipped to detect and/or sweep mines. Some units only have a mine-detection (not sweeping) capability, and will patrol the area attempting to detect mines. Units with minesweeping equipment will patrol their mission area hunting for mines. ROVs embarked on a ship assigned to the mission will automatically be added to it as well.
4. Set the Mission Doctrine/ RoE/ EMCON by pressing the button.
5. Set 1/3 RULE: If checked, will keep 1/3 of the available units on station if possible.
6. Verify the proper reference points are activated and add by pressing the Add Points Currently Highlighted on Map button, which populates the Edit Mine Clearing Area reference point list. You can remove any by selecting reference points in this list and pressing the remove selected button.

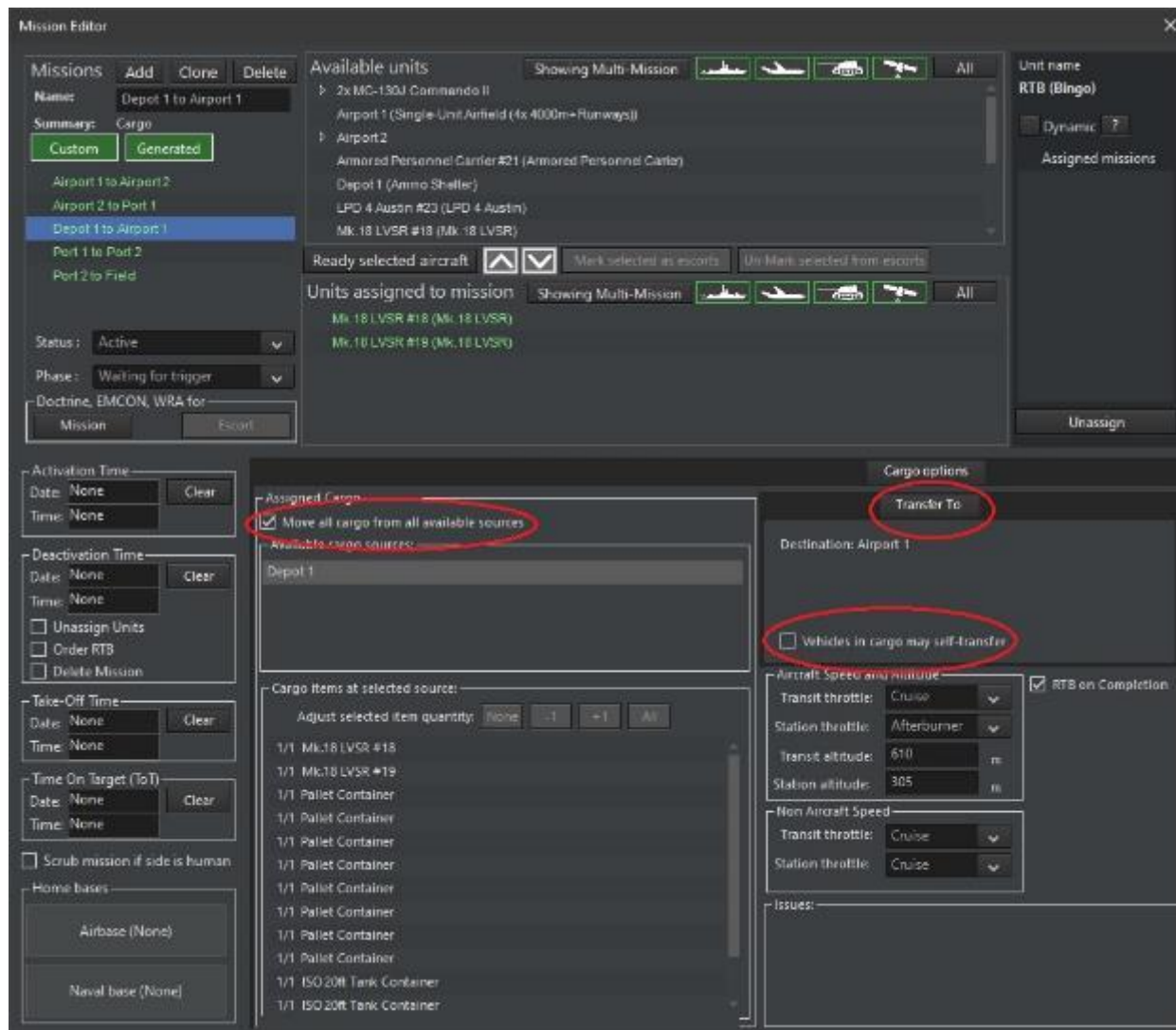
*Note: For long-term mine-clearing missions, it's important for the scenario designer to have a port and the player to have a rotating array of minehunters under the 1/3rd Rule. Mine removal vessels are **expected** to be damaged from the blast of the mines, and so having a rotating cycle of sweepers leaving when repaired and returning when damaged is both realistic and efficient.*

7.2.7 Cargo Mission

Cargo missions should be used for high-unit amphibious or airborne insertions, although the player can still manually unload if they desire. For the AI, it is a necessity.

Historical Note: Despite, or perhaps because of, the inherent risks and complexity of amphibious operations, the overwhelming majority of them after 1900 have not only succeeded but succeeded with the defenders

suffering far more than the attackers. The most likely explanation is that the would-be attackers know this, allocate disproportionate resources, plan extra-carefully, and only launch one if they know it will work.



There are two types of cargo missions: Delivery and Transfer.

Delivery cargo missions work as before, with the destination being a zone defined by RPs on the map. This type of mission will unload cargo into action / for use in the simulation (units will unload from cargo onto the map where they can be given orders, etc. Ammo and fuel will move into unit fuel records / magazines.)

Transfer cargo missions are for moving cargo from one holding unit to another –from airbase to airbase, port to port, or supply facility to airbase, etc. The cargo is not unloaded onto the map but moved from the starting cargo source into one of the transporting units assigned to the mission, transported to the destination, and then moved from the transporting unit into destination unit's cargo.

Creating a Transfer cargo mission uses the same procedure as creating a ferry mission – select a destination unit (rather than a group of reference points) before you create the new mission.

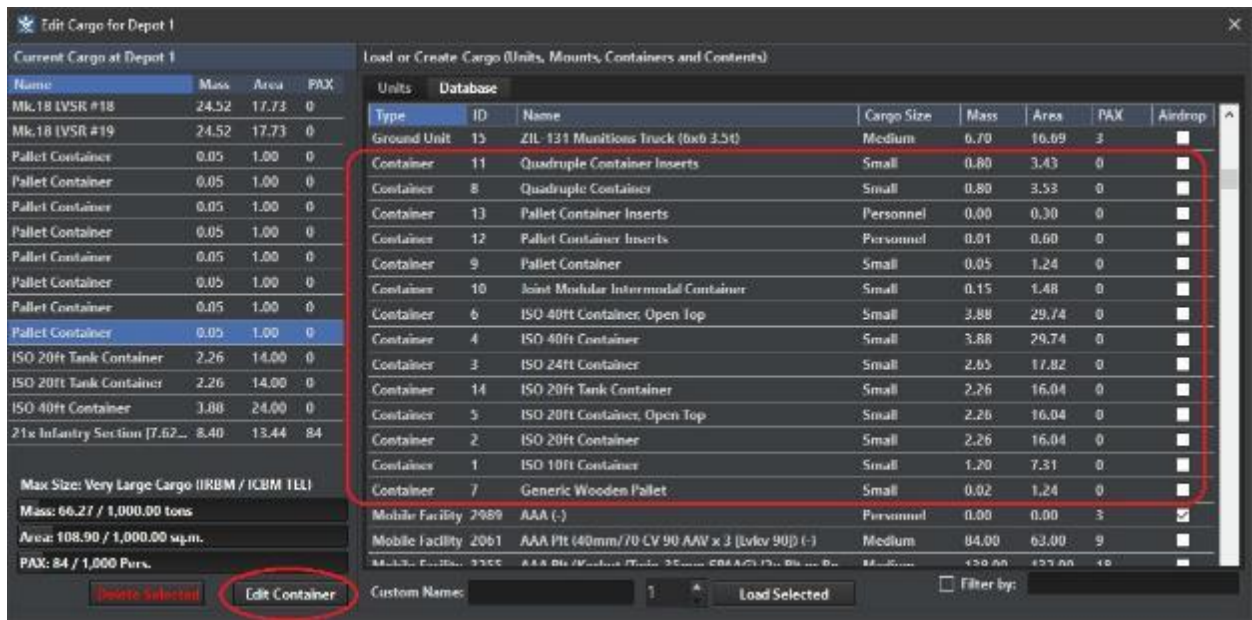
Ships and aircraft transfer or deliver cargo from their starting host unit. Ground units assigned to a cargo missions should start the scenario loaded as cargo within the source unit (i.e. a ship, fixed facility, airbase, etc.)

They will automatically exit cargo and appear on the map as soon as the mission activates and they are within range of the mission destination.

Move All Cargo From All Available Sources: This is a new option for any cargo mission to move all the cargo from the available source(s). You can click this checkbox instead of having to manually go through and assign every possible cargo item. It also allows cargo to ‘flow’ through a system of cargo missions without the sim needing to know what cargo is / is not going to be transferred to a cargo source by some other cargo mission.

Vehicles Stored in Cargo May Self-Transfer: This is a new option found on the ‘Transfer To’ tab for cargo transfer missions. Setting this option ‘on’ allows ground units (not mobile facilities or other cargo) that are assigned to the cargo mission to transport themselves from the starting cargo source to the destination unit’s cargo. They will exit cargo onto the map, travel to the destination unit, and then enter the cargo of the destination unit. In order for this to work the ground unit needs to be in the mission source’s cargo, assigned to the mission, AND in the list of cargo to be transported (or you have the ‘move all’ option set to on.)
(Blank space below due to graphic on next page.)

7.2.8 Cargo / Edit Cargo



New cargo type ‘container’ is now available (NOTE! DB3K 493 or later is required.) Cargo containers are added/removed from cargo the same as other types of cargo. Once a container has been added to cargo you can select it from the current cargo list on the left and click ‘Edit Container’ to put other cargo inside the selected container.

Cargo containers can contain ammunition, fuel, or user-defined contents. User-defined contents can be assigned a name, size, and mass via the Edit Container form. Containers are loaded and moved in the same manner as other cargo types but can’t be unloaded directly to the map as independent entities – they are always in cargo.

If cargo containers are moved as part of a cargo delivery mission they will be delivered into nearby (within 2nm) existing supply-type facilities if possible, or a new 'forward arming and refueling point' facility will be created to hold the containers if none are available.

Fuel in cargo containers that is delivered via a cargo delivery mission will be added as available fuel of the destination facility. This fuel can then be used by other units to refuel.

Ammunition in cargo containers that is delivered via a cargo delivery mission will go into the magazine of the destination facility. These can be used to rearm other units that use the same ammunition type (NOTE! Check the DB ID number to confirm your cargo ammunition matches the type used by the unit you want to re-supply.)

User-Defined Cargo: This is a catch-all type of cargo for items the user wishes to move and track as cargo but which have no effect within the Command simulation. User-defined cargo can be used to track the movement and delivery of items like food, medical supplies, spare parts, etc.

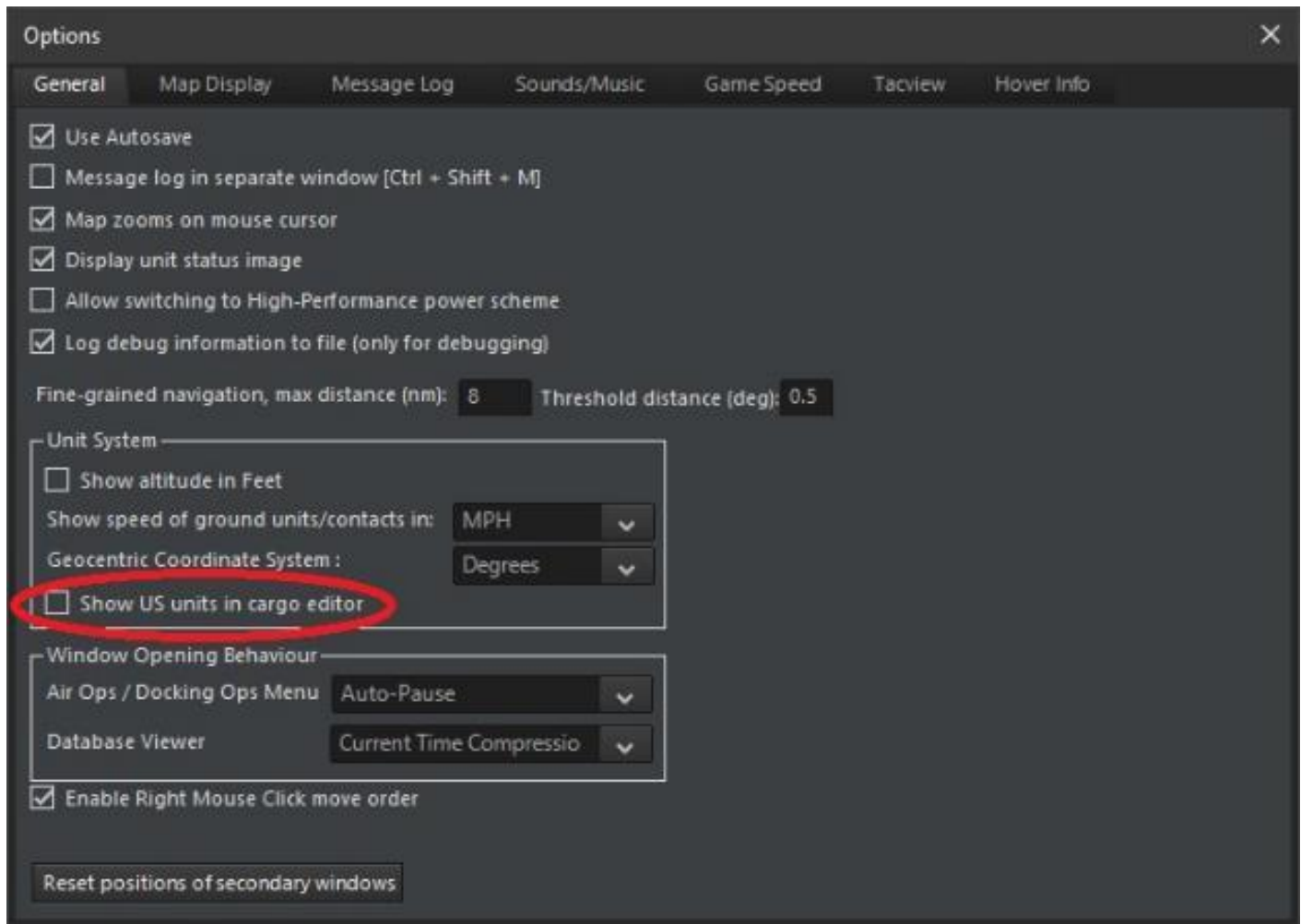
7.2.8.1 Creating a Sequence of Cargo Missions

A single cargo mission can handle movement of cargo from the assigned source(s) to one destination. By using the 'Move All Cargo' option the user can create a sequence of cargo mission to ultimately move cargo through a series of intermediate destinations and on to delivery in the field. The 'Move All Cargo' option is required for this sort of system as individual cargo missions do not know what cargo may be delivered to their cargo source(s) by other cargo missions.

Implementing a full cargo routing and management system was outside the scope of this development task, so the ability to use different missions to route different cargo items from the same source to different destinations is limited to 'originating' cargo sources – i.e those where you know no other cargo mission will be delivering cargo TO the cargo source.

(Blank space below due to graphic on next page.)

7.2.8.2 Units of Measure in Cargo Editor



This option will change the display values show in the Edit Cargo and Edit Container forms to use US units of measure (feet / square feet / tons / pounds / gallons) rather than metric units. This option applies ONLY to those two forms. The forms will refresh automatically if change the option.

7.3 Reference Points

Reference points are markers that can be placed on the map. Reference points are used to define mission areas, objectives, and marking locations.

Placing Reference Points

You can place reference points on the game display in several different ways:

- Pressing the Ctrl button on the keyboard and right-clicking the mouse will give you two options that give that allow you to quickly add reference points.
- Add reference point: Selecting this will let you select a point on the map to drop a reference point.
- Define area: Select this will let you left click drag out an area defined by four reference points. This is very useful in creating square areas for missions. Defined areas may be square or circular.
- You can also place a reference point from the menu by selecting add reference point from the Mission + Ref Points drop down menu.

Selecting and Deselecting Reference Points

To select and deselect reference points just click on them on the UI. With no special properties they appear as gold diamonds when selected and dim x's when not. You can click and drag to select more than one reference point.

Deleting Reference Points: To delete a reference point, select the reference point and select Delete Selected Ref Points from the Missions and Ref. Points drop-down menu.

Moving Reference Points: If unlocked, you can move reference points by clicking and dragging them on the map.

Naming Reference Points: By default, reference points are given arbitrary names based on unit count when added. To rename the reference point make sure it is the only reference point selected and press the "R" button on your keyboard. The Rename Reference dialog should appear allowing you to change it. If it does not, you likely have another reference point selected.

7.3.1 Changing Reference Point Properties

You can change several properties of reference points in the game in the Mission and Ref. Points drop-down.

Making Reference Points Relative: These properties give you the ability to anchor reference points to moving surface unit of your or a friendly side. These are set by selecting reference points, choosing the property in the drop-down list and then selecting the unit or group in the display they will be relative to. This allows you to create mission(s) with defined areas that move relative to a certain unit or its course, such as assigning any aircraft to a mission that protects a moving surface unit (AAW, ASW) or units.

Make selected Reference Points Relative (Fixed) bearing to... : Reference points set to this maintain their position relative to the selected unit or group. This is useful when you know the bearing of a known threat and want to keep a patrol between it and the selected unit regardless of the unit's course. It is also useful for missions where staying on station is important. Reference points that are set to this always have an [f] in their name.

Make selected Reference Point(s) relative (Rotating) bearing to ... : Reference points set to this maintain their position relative to the selected unit or group's course. This is useful when you would like to patrol ahead of a group or unit regardless of its course. Reference points that are set to this always have an [r] in their names.

Locking and Unlocking Reference Points: You can lock and unlock reference points in the game by selecting the reference points and choosing the Lock and Unlock Ref. Point(s) selections in the Mission + Ref Points drop down menu. Locked reference points appear as padlocks are unchangeable by players in game mode.

Delete Selected: Deletes any selected reference point.

De-Select All Reference Points: Deselects any currently highlighted reference points. Can also be made by selecting CTRL-END

7.4 Operations Planning

To fully comprehend this function, it is essential to have a good knowledge of the mission editor in general, and cargo missions in particular. Previously, in Command, a given unit could only be assigned to a single mission. If you wanted to assign the unit to another mission, you would have to manually unassign it from the current mission and then assign it to a new one.

7.4.1 Nomenclature to understand

Triggers

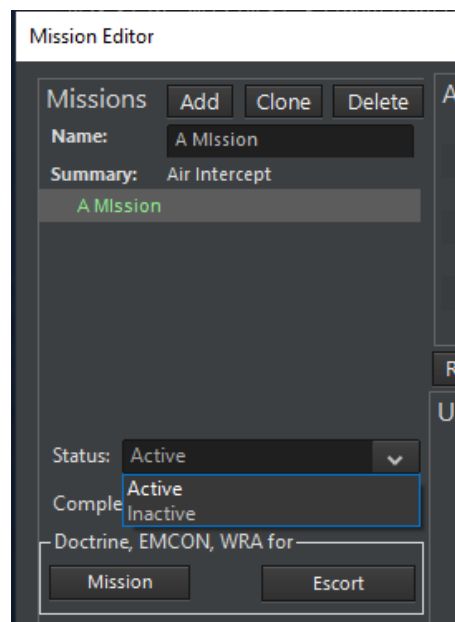
A trigger in the operation planner is a condition that gets checked each simulated second. If the conditions are met the trigger will execute a specific action. There are two possible actions: start a mission or tag a mission as satisfied.

Mission status

A mission's status in Command can be either Active or Inactive.

Active means the mission is evaluated by the Command simulation, but it doesn't necessarily mean that the mission serves a purpose or has units assigned to execute it.

Inactive means that the mission is completely ignored by the simulation until it becomes Active.

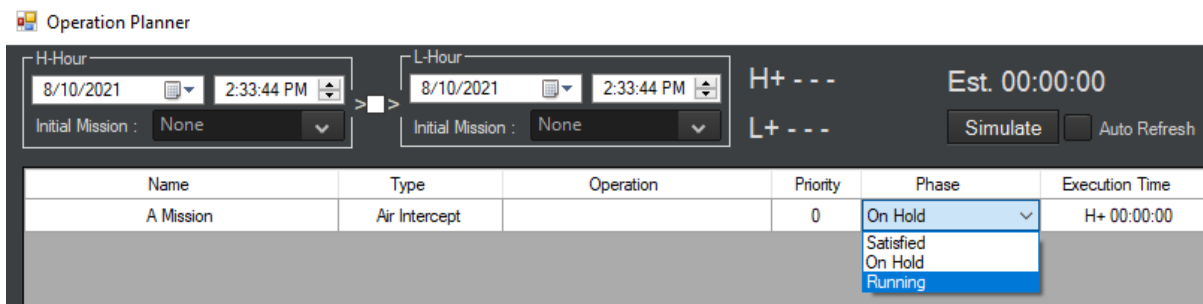


IMPORTANT: It is strongly recommended to leave all missions active, especially when working with the operation planner. Set mission as inactive only for draft missions or manually controlled missions.

Mission phases

A mission phase is a new concept introduced with the operation planner and is not related to the existing mission status.

On Hold:



The mission has not yet started, the units assigned to the mission won't execute the mission.

Satisfied: The mission is considered to have achieved enough of its objective for assigned units to consider other missions, but a satisfied mission does not necessarily end. This tag is used for mission triggers and for multi-mission priority.

Running: A running mission will have its assigned mission to execute the mission.

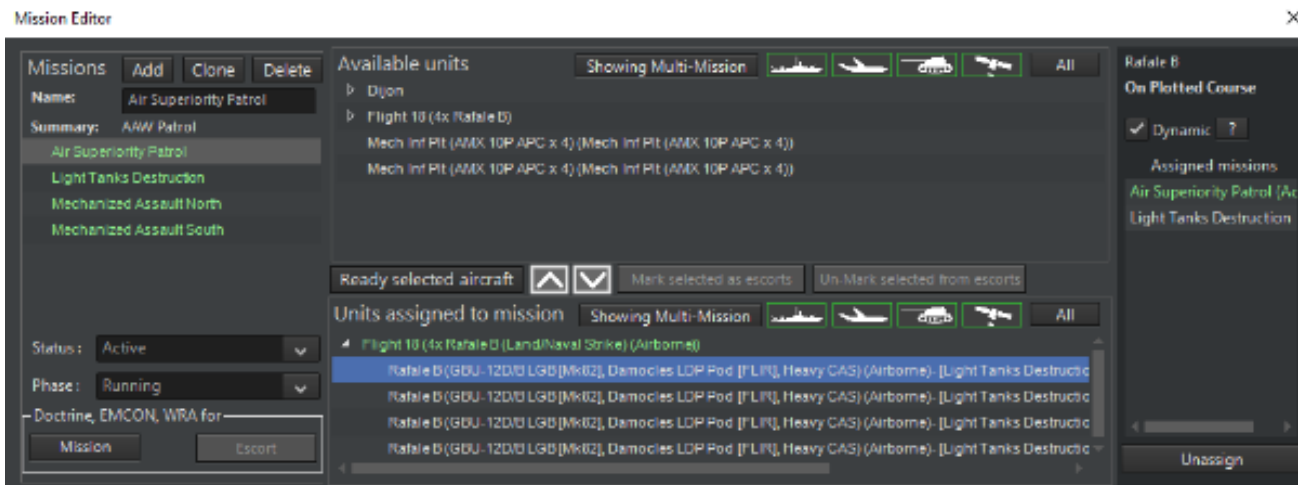
H-Hour: In Command, H-Hour designates the date and time at which the mission designated as the initial mission for H-Hour is to start. H-Hour in Command isn't strictly an H-hour according military terminology as it can be customized without restriction.

L-Hour: In Command, L-Hour designates the date and time at which the mission designated as the initial mission for L-Hour is to start. L-Hour in Command isn't strictly an L-hour according military terminology as it can be customized without restriction.

7.4.2 Understanding the concept of dynamic units

The operations planner provides the capability for a single unit to be assigned to multiple missions. Of course, the unit can only execute one mission at a time, but you can now prioritize which mission it should execute. Mission priority is set via the operation planner.

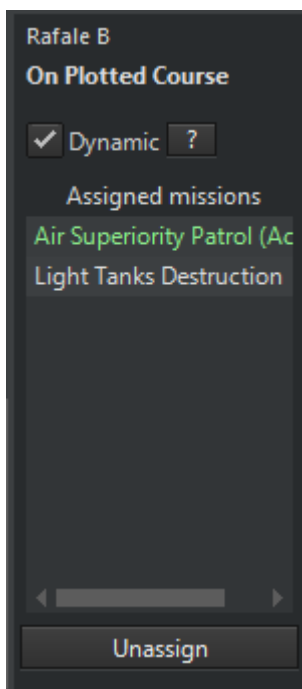
Let's assume we have a scenario where a group of aircrafts is assigned to a patrol mission. These aircrafts should patrol an area from a given time and then start a strike mission against a group of tanks. Without the operation planner, the player would need to track the time and then manually switch each aircraft to a new mission, at the right moment. Thanks to the operation planner such behavior can be automated, and in more complex environment, our units could even behave like a reactive AI, aware of the simulation at the strategical level.



IMPORTANT ! A unit without “Dynamic” Checked will be ignored by the operation planner dynamic assignment, if you want a unit to work with operation planner’s feature you **MUST** designated it as a dynamic unit.

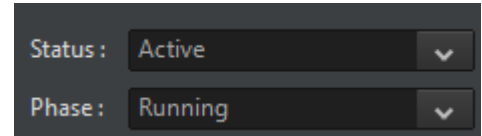
Notice that since we have toggle “Showing Multi-Mission” we are now allowed to assign multiple mission to a unit, they are still shown in the “Available units” list despite having an assigned mission.

This panel on the right side of the mission editor shows all missions assigned to this unit. The mission in green in the one currently executed by the unit. Missions don’t have an order of execution, but a priority, which is set in the operation planner, as we will see later.



The “Dynamic” checkbox allows the unit to be assigned to multiple missions, units are all unchecked by default to reflect default command behavior.

Below the mission list we can see the current status and phase of the mission. We will see later in the operation planner chapter as “Phase” is a new way of managing the mission dynamically and is closely related to multi-mission.



A screenshot of a user interface showing two dropdown menus. The first menu is labeled 'Status:' and has 'Active' selected. The second menu is labeled 'Phase:' and has 'Running' selected. Both menus have a downward arrow icon on the right side.

It is on this screen that you decide to add the unit to all of the missions you want it assigned to. At this point you don’t have worry about priority or to select a current mission as the operation planner will manage this for you.

In above’s example we see that Rafale B is assigned to both the “Air Superiority Patrol” and “Light Tanks Destruction” missions, and the active mission for this unit is the one in green: “Air Superiority Patrol.” Depending on the configuration of the operation planer, this unit may automatically switch to the “Light Tanks Destruction” mission at some point in the future.

Don’t worry if the multi-mission mechanism is not entirely clear to you yet, as the mission editor is only half the story. The next chapter on operation planner will explain the other half.

7.4.3 Mobile facility vs Ground unit and Split/Merge ground units

One of the most significant capabilities brought by the operation planner, is the possibility to have units dynamically change mission. But not all types of units are eligible for this kind of behavior. As part of that, It is important to understand the core difference between:

- **Ground units** as facility which is a legacy implementation where ground unit are assimilated as moving “facility”
- **Mobile facilities**, a new implementation, which works like others active units such as aircraft, ships etc...

The first hold a group of units abstractly represented as “mounts”, while the last is a fully simulated individual unit. The core difference that interests us here is that ground units as facility are transported as cargo which doesn’t have an existence in the simulation until it has landed (and spawned).

This means that we can’t assign or queue them a mission until they have landed, and you will not be able to achieve a fully autonomous behavior for your units, in this case.

Since the 8th November 2021 update we allow cargo operation for active units, meaning that all these limitation are now removed. However, you must have the right methodology to achieve this.

Unless you have to achieve backward compatibility or if a database entry is missing, **you MUST use ground units NOT mobiles facilities** to use the operation planner at its fullest.

On the new cargo edition form, note the “type”, at the moment, the database has more content for mobile facility, fortunately, our database is expanding to have the active ground units catch up in content with the rest.

Edit Cargo for LHA 1 Tarawa #2

Current Cargo			
Name	Mass	Area	Crew
14x LAW-25 (25mm/75 M34...	182.00	224.00	126
12x AAW-P7/A1	360.00	324.00	230
8x 155mm/39 M777 Towed ...	26.40	148.80	48
3x 81mm Mortar	2.40	3.04	24
8x BGM-71A TDW Section	3.20	5.12	32
3x FGM-140 Javelin	2.40	3.04	24
66x Vehicle (Truck, HMMWV)	189.00	630.00	378
35x Vehicle (Truck, Unarmed)	200.00	420.00	700

Max Size Large Cargo (Tank, TEL, Trailer)
 Mass: 1,045.40 / 5988 tons
 Area: 1,759.60 / 3130 sq.m.
 Crew: 1620 / 1903

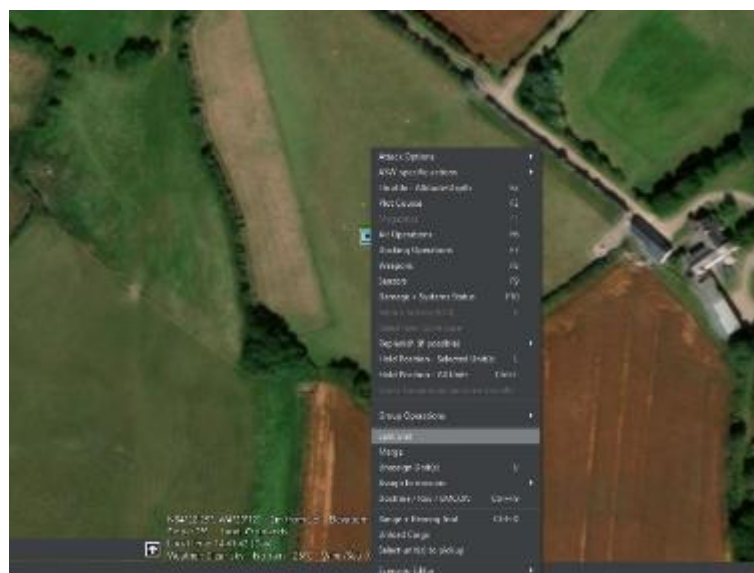
1 Unload Selected

Available Cargo						
Type	ID	Name	Cargo Size	Mass	Area	Crew
Ground Unit	8	Fireteam Leader (5.56mm AR, RCO 4x optic, 40mm GL)	Personnel	0.10	0.16	0
Ground Unit	4	LAW-25 (Light Armored Vehicle, Mech. Recon)	Medium	12.30	15.90	3
Ground Unit	6	LAW-AD (Light Armored Vehicle, Air Defense)	Medium	12.80	15.98	3
Ground Unit	5	LAW-AT (Light Armored Vehicle, Anti-Tank)	Medium	12.30	15.90	3
Ground Unit	22	M1A1 Abrams MBT (Standard Armor)	Large	57.00	29.02	4
Ground Unit	23	M1A1 (i) Abrams MBT (i) heavy Armour variant)	Large	57.00	29.02	4
Ground Unit	17	M3 (Zachlem) (Halftrack APC)	Medium	9.00	13.72	1
Ground Unit	16	M3 (Zachlem) TCM-20 (i) Halftrack, 20mm cannons)	Medium	9.00	13.72	3
Ground Unit	14	Radar (Straight Flush (1591)) (Assoc. w/ SA-6)	Large	34.00	23.48	3
Ground Unit	7	Rifleman (5.56mm AR, RCO 4x optic)	Personnel	0.10	0.16	0
Ground Unit	13	TEL (SA-6a Gainful (2P25)) (1st Bn + Long Track + Thin ...)	Large	34.00	23.48	3
Ground Unit	15	ZIL-131 Munitions Truck (6x6 3.5t)	Medium	6.70	16.69	3
Mobile Facility	2989	AAA (-)	Personnel	0.00	0.00	3
Mobile Facility	2051	AAA Pht (40mm/70 CV 90 AAW x 3 [Lvlev 90]) (-)	Medium	84.00	63.00	9
Mobile Facility	3255	AAA Pht (Korkut (Twin 35mm SPAAG) (2x Pht pr Bn, 4x ...)	Medium	138.00	132.00	18
Mobile Facility	2720	AAA Pht (LD-2000 x 2) (3x pr Bty, Used with HQ-64)	Large	48.00	46.00	8
Mobile Facility	2107	AAA Pht/2 (23mm ZSU-23-2 BTR-ZD x 2) (2x pr Pht)	Medium	16.00	24.00	6
Mobile Facility	2108	AAA Pht/2 (23mm ZSU-23-2 BTR-ZD x 2) (2x pr Pht)	Medium	16.00	24.00	6
Mobile Facility	80	AAA Pht/2 (23mm ZSU-23-4 Shilka x 2) (2x pr Bn)	Large	38.00	38.00	8
Mobile Facility	908	AAA Pht/2 (23mm ZSU-23-4 Shilka x 2) (2x pr Bn)	Large	38.00	38.00	8
Mobile Facility	2748	AAA Pht/2 (23mm ZSU-23-4 Shilka x 2) (2x pr Bn)	Large	38.00	38.00	8

Custom Name: 1 Load Selected

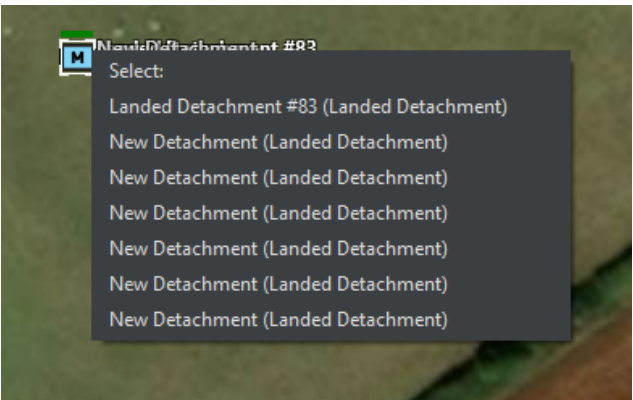
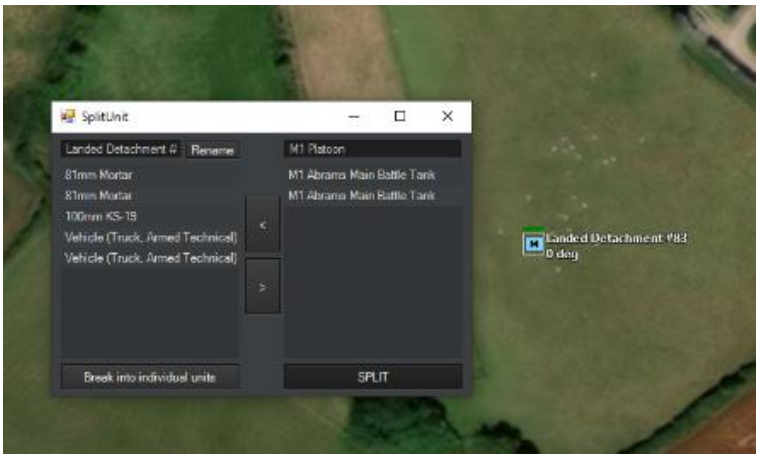
Ability to Split/Merge ground units (facilities)

Legacy ground units associated as “facilities” can now be rearranged through this new tool.



Select an eligible unit, such as a landed detachment, right click on it to bring the context menu and click on “Split unit”.

This brings you this menu, with the details of the detachment. You can also break the detachment into individual units with a single click



Two eligible units can be merge together:



7.5 Operation Planner

The operation planner adds a new level of interaction between missions and allows units to be dynamically reassigned from one mission to another.

Since the operation planner is sometimes tied to a landing plan, we have here the concepts of H-Hour and L-Hour to indicate overall operational time. These can be of course be ignored, or used in a different purpose.

On the top left, are where you define the parameters for the H-Hour and the L-Hour.

H-Hour

8/10/2021 2:33:44 PM

Initial Mission : None

> ☐ >

L-Hour

8/10/2021 2:33:44 PM

Initial Mission : - Master Landing

The H-Hour box on the left is where we can define the date and time to start the H-Hour mission. The H-Hour mission is the initial mission in the operation. L-Hour box works identically to H-Hour but they are independent.

Once an H-Hour or L-Hour is hit the respective initial mission can no longer be changed.

The checkbox in the middle ties the H-Hour to the L-Hour, meaning that the time separation will be constant between H-Hour and L-Hour when you modify the time for either. In other words, H-hour and L-hour will match at all times if that center check box is checked.

Bulk action	Name	Type	Description	Priority	Phase	Execution Time	Activated
☐	Shore A LCAC 1 #4 Wave 1	Cargo	LCAC 1 #4 - Serials #0 #5	10	Triggered	H+ 00:00:00	☒
☐	Shore B LCAC 1 #4 Wave 1	Cargo (Landing plan)	LCAC 1 #4 - Serials #0 #40	40	Waiting for trigger	H+ 00:00:00	☒
☐	Shore A LCAC 1 #4 Wave 2	Cargo (Landing plan)	LCAC 1 #4 - Serials #15 #30	39	Waiting for trigger	H+ 00:00:00	☒
☐	Shore C LCAC 1 #4 Wave 3	Cargo (Landing plan)	LCAC 1 #4 - Serials #55	38	Waiting for trigger	H+ 00:00:00	☒
☐	Shore B LCAC 1 #4 Wave 4	Cargo (Landing plan)	LCAC 1 #4 - Serials #70	37	Waiting for trigger	H+ 00:00:00	☒
☐	Shore A LCU 1646 #2 Wave 1	Cargo (Landing plan)	LCU 1646 #2 - Serials #-1	30	Waiting for trigger	H+ 00:00:00	☒
☐	Shore B LCU 1646 #2 Wave 2	Cargo (Landing plan)	LCU 1646 #2 - Serials #25	29	Waiting for trigger	H+ 00:00:00	☒
☐	Shore B LCU 1646 #2 Wave 3	Cargo (Landing plan)	LCU 1646 #2 - Serials #60	28	Waiting for trigger	H+ 00:00:00	☒
☐	Shore A LCU 1646 #3 Wave 1	Cargo (Landing plan)	LCU 1646 #3 - Serials #5 #10	30	Waiting for trigger	H+ 00:00:00	☒
☐	Shore C LCU 1646 #3 Wave 2	Cargo (Landing plan)	LCU 1646 #3 - Serials #50	29	Waiting for trigger	H+ 00:00:00	☒
☐	Shore A LCU 1646 #3 Wave 3	Cargo (Landing plan)	LCU 1646 #3 - Serials #65	28	Waiting for trigger	H+ 00:00:00	☒
☐	Inland B Green Pawn #2 Wave 1	Cargo (Landing plan)	Green Pawn #2 - Serials #35	10	Waiting for trigger	H+ 00:00:00	☒
☐	Inland A Green Pawn #12 Wave 1	Cargo (Landing plan)	Green Pawn #12 - Serials #45 #75	10	Waiting for trigger	H+ 00:00:00	☒
☐	Inland B Champion #2 Wave 1	Cargo (Landing plan)	Champion #2 - Serials #85	10	Waiting for trigger	H+ 00:00:00	☒

The spreadsheet in the center lists all the side’s missions. Most columns are informational. Generally, you will be dealing with 2 columns: **Priority** and **Phase**.

7.5.1 Mission priority

The priority of a mission does NOT designate its supposed position in a mission execution queue. It indicates to its assigned units how important this mission is **at this moment**. The mission priority is used by units assigned to multiple missions to decide which mission to execute at any given time. A unit having a mission in “On Hold” phase will not have this mission evaluated for the active mission evaluation.

7.5.2 Mission phases

Operation Planner

H-Hour: 10/08/2021 14:33:44 L-Hour: 10/08/2021 14:33:44 Initial Mission: None

Simulation: H+ --- L+ --- Est. 00:00:00 Simulate Auto Refresh

Filter: Go Sort order: Estimated Execution Time

Bulk action	Name	Type	Description	Priority	Phase	Execution Time	Activated
☐	Shore A LCAC 1 #4 Wave 1	Cargo	LCAC 1 #4 - Serials #0 #5	10	Triggered	H+ 00:00:00	☒
☐	Shore B LCAC 1 #4 Wave 1	Cargo (Landing plan)	LCAC 1 #4 - Serials #0 #40	40	Waiting for trigger	H+ 00:00:00	☒
☐	Shore A LCAC 1 #4 Wave 2	Cargo (Landing plan)	LCAC 1 #4 - Serials #15 #30	39	Triggered	H+ 00:00:00	☒
☐	Shore C LCAC 1 #4 Wave 3	Cargo (Landing plan)	LCAC 1 #4 - Serials #55	38	Satisfied	H+ 00:00:00	☒
☐	Shore B LCAC 1 #4 Wave 4	Cargo (Landing plan)	LCAC 1 #4 - Serials #70	37	Waiting for trigger	H+ 00:00:00	☒
☐	Shore A LCU 1646 #2 Wave 1	Cargo (Landing plan)	LCU 1646 #2 - Serials #-1	30	Waiting for trigger	H+ 00:00:00	☒
☐	Shore B LCU 1646 #2 Wave 2	Cargo (Landing plan)	LCU 1646 #2 - Serials #25	29	Waiting for trigger	H+ 00:00:00	☒

A mission phase is a new concept introduced with the operation planner extension, it is not related to the mission status and should not be mistaken with it. Active missions can still be running to completion even if it cycles through all phases. An active mission only stops upon destruction of targets, end of a single patrol, units destroyed, etc. On the other hand, a mission can satisfy its objectives relative to the Ops Planner without the mission ending. Phases are used only in relation to the Ops Planner.

Waiting for trigger: The mission may (or may not) have already started yet but queued units won't evaluate this mission when choosing one to be assigned to. No dynamic units will be assigned to a mission "Waiting for Trigger".

Satisfied: The mission is considered to have achieved enough of its objective for the Ops Planner to assign current assigned dynamic units to other missions. Keep in mind that a satisfied mission does not mean its ending. Units currently operating in an active mission will continue to execute that mission. But dynamic units waiting to take off or execute the current mission might potentially be assigned to new Triggered missions, depending on priorities. The Satisfied tag is used for mission triggers and for multi-mission priority in the Ops Planner only.

Triggered : A running mission is the only way the Ops Planner evaluate a mission as a valid candidate for multi-mission units assigned to it. Therefore, a queued unit that is in a Waiting for Trigger or Satisfied mission may become assigned to it. The Ops Planner will only check for dynamic units to be assigned to a mission if its triggered through the Ops Planner.

The Ops Planner evaluates units assigned to multiple missions for assignment to the most appropriate Triggered mission, based on the priority. A unit that is active in a mission in "Satisfied" phase will still evaluate the satisfied mission and may continue that mission if no other missions are available.

7.5.3 Column Definitions

Bulk actions

Bulk Actions

Check All	On checkmarked missions >	Delete	Unassign all units	Unassign all queued units	Activate	Desactivate	Satisfied
Uncheck All							

Bulk action tools are located on the bottom of the operation planner. These are used to select multiple mission at once and perform simultaneous modifications on them.

A dark-themed UI element consisting of a rectangular box. On the left, the word "Filter" is written in a light color. To its right is a text input field with a light gray border. Further right is a button labeled "Go" in a light color.

Name

Just like for operation, the mission's name helps for organization. However use made Lua scripts might reference a mission by its name and it is recommended to be careful when changing mission's name in such situation.

Type

This is the mission sub type as defined in Command's simulation.

Description

This is purely an informational tag and does not affect the simulation at the moment. Use and edit this field to organize yourself.

Generated mission will usually contain some generated information

Execution Time

The time, relative to H-Hour at which the mission is estimated to begin. See the chapter "Working with estimation" for more details.

Filter

The filter tool is used to do a specific term research on mission and filter the result of this search query.

7.5.4 Triggers

This chapter is the core of the operation planner capabilities. It allows relationships between mission and a dynamic approach when executing missions.

A trigger is like a set of a condition and an action, if the condition is met, then we do an action.

Command evaluates all these triggers every second.

For mission, there are 2 types of actions:

- We start a mission (we set the mission's phase as "Running").
- We finish a mission (we set the mission's phase as "Satisfied").

It is important to understand that Command doesn't have a concept of mission completion, when tagging a mission as satisfied, command only indicates that the mission fulfilled enough of its objective to allow its assigned units to evaluate other mission assignment options. Of course all relevant missions have already implicit mission completion mechanisms, a cargo mission will stop operation once the task is done, a strike mission won't launch again to strike a nonexistent targets , etc...

These 2 actions are tied to a set of conditions.

Starting a mission with triggers

Select any of the mission and look on the right side of the operation planner window. Notice the main block called “Triggers to Start Mission”, and how it is separated into 3 smaller blocks.

These smaller blocks are individual conditions. Checking them means this specific condition will be evaluated.

The dropdown on the right of each smaller block is called a conditional operator.

As you can see there are 3 types of triggers that can start a missions:

The screenshot shows a window titled "Triggers to Start Mission". It contains three main sections, each with a checkbox and a conditional operator dropdown (OR):

- Time-based trigger:** The checkbox is unchecked. It includes a duration field set to "0" days and "00:00:00" hours/minutes/seconds.
- Mission(s) Satisfied:** The checkbox is checked. It features two lists: "Missions to check" (containing "Air Superiority Patrol") and "Mission Pool" (containing "Mechanized Assault South", "Light Tanks Destruction", and "Mechanized Assault North"). Navigation arrows are between the lists.
- Lua script trigger:** The checkbox is unchecked. It includes an "Edit Script" button and a "Description" field with a "(25 char max)" limit.

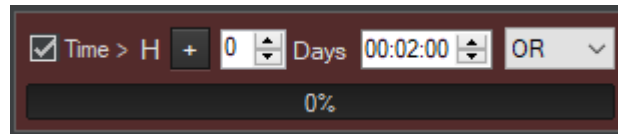
- A time based trigger
- A mission dependency trigger
- A Lua script trigger

Time-based trigger

This is a close-up of the time-based trigger section from the previous screenshot. It shows the checkbox, the duration input fields (0 days, 00:00:00), and the conditional operator dropdown (OR).

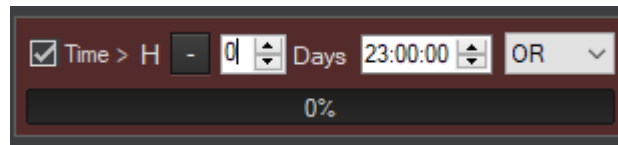
This will be triggered when the scenario date reaches the defined H-Hour plus or minus a given duration.

Exemple #1 : This trigger will be **true** if we reach H+ 2 hours



A trigger configuration window with a dark red background. At the top, there is a checked checkbox labeled 'Time > H', followed by a '+' button, a numeric input field containing '0', a 'Days' label, a time input field containing '00:02:00', and an 'OR' dropdown menu. Below these fields is a progress bar showing '0%'.

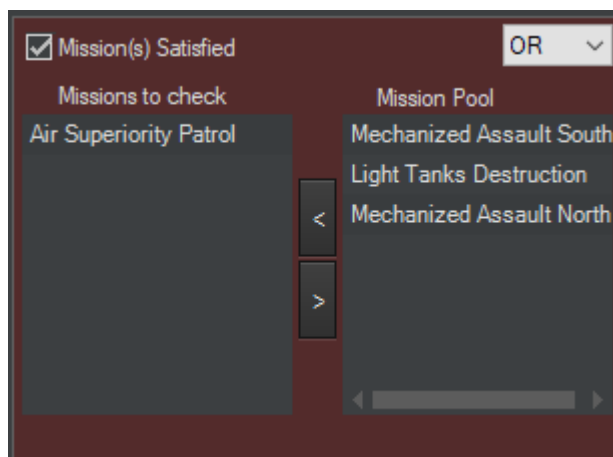
Exemple #2 : This trigger will be **true** if we reach H- 23 hours



A trigger configuration window with a dark red background. At the top, there is a checked checkbox labeled 'Time > H', followed by a '-' button, a numeric input field containing '0', a 'Days' label, a time input field containing '23:00:00', and an 'OR' dropdown menu. Below these fields is a progress bar showing '0%'.

Mission dependency trigger

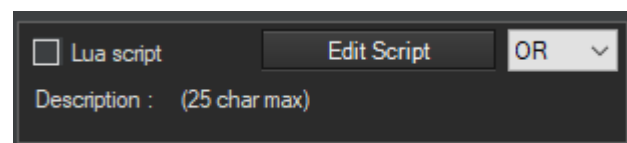
This will be triggered if all missions in “missions to check” are in “Satisfied” phase. In this example, it will be triggered when “Air Superiority Patrol” mission is in “Satisfied” Phase.



A mission dependency trigger configuration window with a dark red background. At the top, there is a checked checkbox labeled 'Mission(s) Satisfied' and an 'OR' dropdown menu. Below this, there are two columns: 'Missions to check' and 'Mission Pool'. The 'Missions to check' column contains a list with 'Air Superiority Patrol'. The 'Mission Pool' column contains a list with 'Mechanized Assault South', 'Light Tanks Destruction', and 'Mechanized Assault North'. Between the two columns are '<' and '>' buttons. At the bottom of the 'Mission Pool' column is a scrollbar.

Lua script trigger

This will be triggered if the Lua script contained returns TRUE as a value. See the “Lua scripting” chapter for more informations.



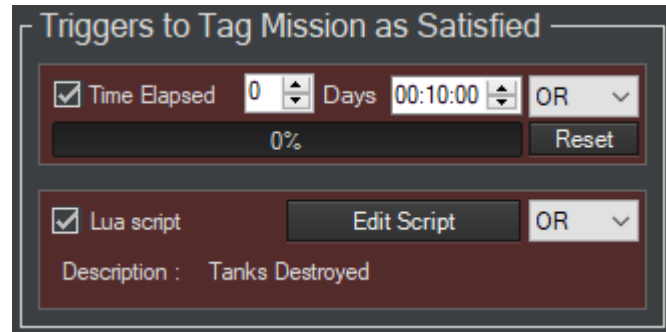
A Lua script trigger configuration window with a dark gray background. At the top, there is an unchecked checkbox labeled 'Lua script', followed by an 'Edit Script' button, and an 'OR' dropdown menu. Below these fields is a text input field labeled 'Description : (25 char max)'.

“Finishing” a mission with triggers

It was mentioned earlier that Command does not have an explicit concept of finished mission.

The triggers to tag a mission as “satisfied” work just like the one to start it.

The first trigger is a time based one, it tracks the elapsed time since the mission’s phase has been set “Running” and will be true once the defined time elapsed.



have
like
to

The second one is a Lua script trigger and work identically to the one in the mission start trigger – it returns the Boolean value of the contained lua script.

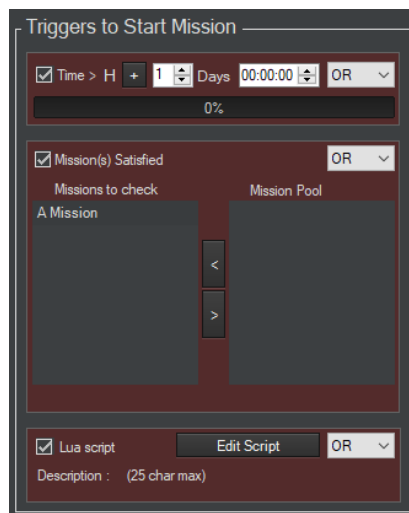
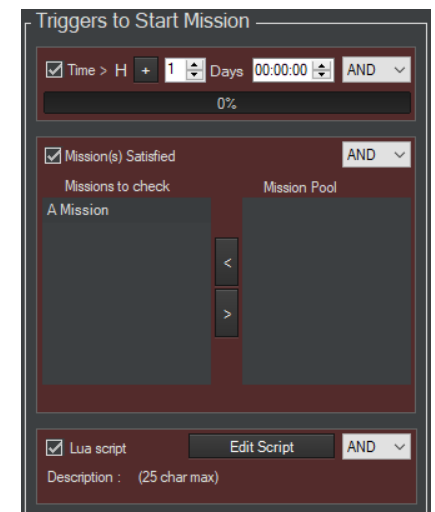
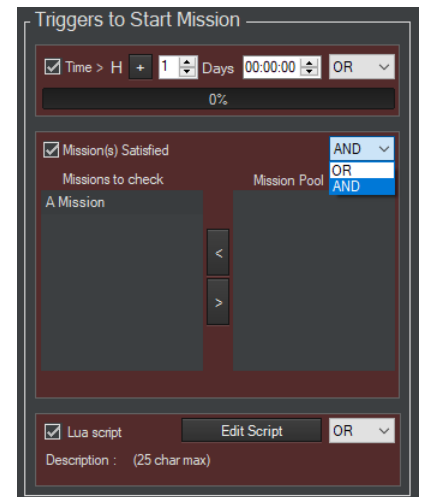
Logical Operators

Triggers, in Command, are tied with logical operators.

Take a look at the pictures below. They represent a set of triggers to start a mission:

We have set all 3 triggers to be checked. On the right of each trigger you can notice a dropdown when you can selected either the OR or AND operator. For the triggers to return TRUE and therefore start the mission (set its phase change to “Running”) each checked trigger is evaluated.

We see in the next example that the first trigger has “OR” operator, the second “AND,” the third “OR.” What it means in this situation is that the second trigger (the one with “AND”) must be true. In addition the “AND” trigger needing to be true, either of the first or third “OR” triggers must also be true. If all conditions are met and the current mission’s phase is “On Hold” then the mission will change it phase to “Running”.



Another example:

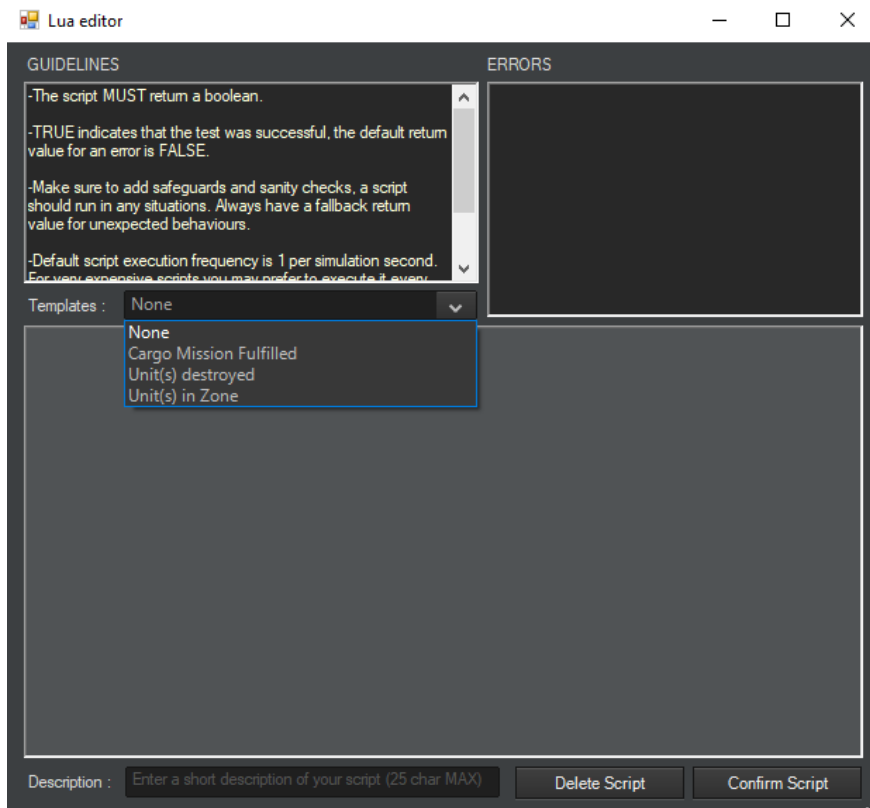
All triggers are checked (and will be evaluated), and each of them have the “AND” operator. This means that the missions will start when all conditions are true.

7.5.2 Operation Planner and Lua Scripting

Command has already a powerful Lua scripting framework. The operation planner allows the integration of your script as a trigger.

Clicking on “Edit Script” in either the “Triggers to Start Mission” or the “Triggers to Tag Mission as Satisfied” group will bring you to an interface where you can add your script in.
Just like the rest of the triggers, the Lua Script trigger will be executed each second for its associated mission. The Lua script must return a Boolean.

Working with estimation



Command is such a complex simulation that giving an accurate estimation of an operation duration could take up to hours of computation. The operation planner provides an instantaneous estimation at the price of reliability.

The operation planner estimation takes into account all the time based triggers and mission dependencies, it simulates a run and then display the estimated execution tie for each mission.

This estimation work only thanks to the user’s input on triggers.

The special case of Lua script

You may have noticed that not all triggers are time-based, some depends on lua script and cannot be reliably predicted. In this case, you will have to manually input a value into the trigger in this trigger, shown on the right. It is not necessary to check (enable) the trigger, having an unchecked “Time Elapsed” trigger with a value basically tells the operation planner : “Only use this trigger when estimating execution time”. In this exemple, we assume the mission will be satisfied after 1 hour.

The purpose of the air tasking order (ATO) is to enable the control of multiple air forces in one operational environment. At the most basic level, it provides an overview of all the air sorties as well as the appropriate status, missions used and various timings as appropriate.

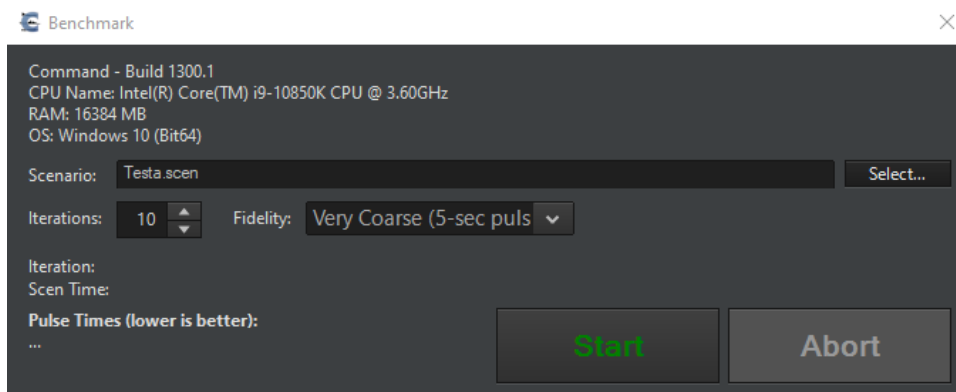
The show flights option at the top of the screen enables you to filter and show the flights that you are more interested in during a given scenario. The show airborne and show planned flightplans option gives you the ability to see the flightplan and make changes to it of the currently selected flight. Note that after selecting any flight, it is possible to click the edit flightplan button at the base of the screen to make direct edits to the flightplan of that group.

The create, copy and delete flights allow you to modify any flight that you have highlighted in the ATO. Creating a blank flight allows you to pre-plan all aspects of a flight without having a set mission in place. You can even design a flightplan and then select Change a/c type and loadout for the purposes of pre-selecting the appropriate aircraft from an existing aircraft that you possess in the scenario already.

It is also possible to change the type of flightplan to a flightplan template. This template can be used to create a standard model with all the appropriate settings built in that can later be applied to other units and missions at a later time. The best use of the ATO is to monitor what a large quantity of aircraft are doing at any given time, and to be able to quickly find everything you need to know about the various missions throughout a scenario.

7.7 Benchmark Mode

The benchmark mode allows you to test the performance of a scenario without having to run the main game GUI. This allows scenario editors a chance to see how their creation will run in general terms. To access this mode, select 'Benchmark' from the file menu and the main window will close and the following dialog box will appear.



The scenario button allows you to choose what scenario you would like to benchmark and can be selected from any saved scenarios in the appropriate folder.

Iterations – Allows you to select how many runs through the entire scenario that the benchmark mode will execute.

Fidelity – This allows you to select how precise the moment-by-moment calculations should be completed. In scenarios where there are very large numbers of units, selecting a high fidelity will result in significantly longer run times.

While running, the benchmark mode will indicate what the current scenario time is and it will show you how long each 'pulse' is taking to execute. A shorter pulse time indicates a faster running scenario. Note that when the benchmark is running, if there is no mission termination or units on any of kind of mission, the benchmark mode will run very quickly but nothing will actually be going on in the scenario giving bad results.

When the benchmark is complete, you will receive a report on the different lengths of pulse times as well as the highest level of time acceleration achieved. You can press the abort key at any time if you need to stop the process.

7.8 Lua Reference

Mission Lua Wrapper

PriorityWeight (Read/Write) [int]

How important this mission is (0 = most important) affects multi-mission units assignment dynamic.

OperationName (Read/Write) [string]

Used to organize mission into groups, it is an informational element shown on the operation planner UI.

Completion (Read/Write) [float]

From 0.0 to 1.0, it designates how complete this mission is. This value is mostly unused in most cases at the moment and should be ignored.

Phase (Read/Write) [ENUM]

The phase as interpreted by the operation planner, this rules the whole dynamic in the operation planner and the assignment of multi-mission units.

MissionPhase ENUM :

None = 999

Auto = 0

Completed = 10

Active = 20

OnHold = 30

MISSION START BY TIME TRIGGER

MissionStartTrigger_Time (Read/Write) [int]

The H +/- time condition for the trigger.

MissionStartTrigger_Time_Enabled (Read/Write) [Bool]

Is the trigger used or not ?

MissionStartTrigger_Time_LastResult (Read) [Bool]

What was the last result of the trigger the last time it has been evaluated ?

MissionStartTrigger_Time_Operator (Read/Write) [Bool]

The trigger's operator. True = 'AND', False = 'OR'

MISSION START BY MISSION COMPLETED

MissionStartTrigger_MissionCompleted (Read/Write) [NLua.LuaTable]

The condition for the trigger, defined as all the missions that need to be completed (phase as "satisfied") for the trigger to check.

Exemple :

```
>> local mission = ScenEdit_GetMission('USA','Shore A LCAC 1 #4 Wave 1')
print(mission.MissionStartTrigger_MissionCompleted)
mission.MissionStartTrigger_MissionCompleted = {"LTACBA-0HMBO9DVSMOPI","LTACBA-0HMBO9DVSMOP9","LTACBA-0HMBO9DVSMOPB"}
print(mission.MissionStartTrigger_MissionCompleted)
{ [1] = 'LTACBA-0HMBO9DVSMOPI' }
{ [1] = 'LTACBA-0HMBO9DVSMOPI', [2] = 'LTACBA-0HMBO9DVSMOP9', [3] = 'LTACBA-0HMBO9DVSMOPB' }
```

MissionStartTrigger_MissionCompleted_Enabled (Read/Write) [Bool]

Is the trigger used or not ?

MissionStartTrigger_MissionCompleted_LastResult (Read) [Bool]

What was the last result of the trigger the last time it has been evaluated ?

MissionStartTrigger_MissionCompleted_Operator (Read/Write) [Bool]

The trigger's operator. True = 'AND', False = 'OR'

MISSION START BY LUA CONDITION

MissionStartTrigger_LUADescription (Read/Write) [String]

The label of the lua script as an indicator of its purpose.

MissionStartTrigger_LUA (Read/Write) [String]

The lua script to be executed for this trigger conditions, must return a boolean as a condition result.

MissionStartTrigger_LUA_Enabled (Read/Write) [Bool]

Is the trigger used or not ?

MissionStartTrigger_LUA_LastResult (Read) [Bool]

What was the last result of the trigger the last time it has been evaluated ?

MissionStartTrigger_LUA_Operator (Read/Write) [Bool]

The trigger's operator. True = 'AND', False = 'OR'

MISSION COMPLETED ("SATISFIED") BY ELAPSED TIME OF MISSION EXECUTION

MissionCompletedTrigger_ElapsedTime (Read/Write) [int]

The elapsed time since mission execution for the trigger's condition to return true.

MissionCompletedTrigger_ElapsedTime_Current (Read/Write) [int]

The current elapsed time.

MissionCompletedTrigger_ElapsedTime_Enabled (Read/Write) [Bool]

Is the trigger used or not ?

MissionCompletedTrigger_ElapsedTime_LastResult (Read) [Bool]

What was the last result of the trigger the last time it has been evaluated ?

MissionCompletedTrigger_ElapsedTime_Operator (Read/Write) [Bool]

The trigger's operator. True = 'AND', False = 'OR'

MISSION COMPLETED ("SATISFIED") BY LUA CONDITION

MissionCompletedTrigger_LUA_Description (Read/Write) [String]

The label of the lua script as an indicator of its purpose.

MissionCompletedTrigger_LUA (Read/Write) [String]

The lua script to be executed for this trigger conditions, must return a boolean as a condition result.

MissionCompletedTrigger_LUA_Enabled (Read/Write) [Bool]

Is the trigger used or not ?

MissionCompletedTrigger_LUA_LastResult (Read) [Bool]

What was the last result of the trigger the last time it has been evaluated ?

MissionCompletedTrigger_LUA_Operator (Read/Write) [Bool]

The trigger's operator. True = 'AND', False = 'OR'

Operation Lua Wrapper

The 'Operation' object can be fetched from the side object. There is only one operation object per side.

LHourMission (Read/Write) [LuaWrapper_Mission]

The mission designated to start at L-Hour. As soon as the L-Hour is reached, the mission will change phase to "Running".

HHourMission (Read/Write) [LuaWrapper_Mission]

The mission designated to start at H-Hour. As soon as the H-Hour is reached, the mission will change phase to "Running".

HHour (Read/Write) [String]

The time set as H-Hour.

LHour (Read/Write) [String]

The time set as L-Hour.

HHourEffectiveStartTime (Read/Write) [String]

The effective time at which the H-Hour started counting. Sometimes the actual H-Hour can be different from the one planned initially, or in the past.

LHourEffectiveStartTime (Read/Write) [String]

The effective time at which the L-Hour started counting. Sometimes the actual L-Hour can be different from the one planned initially, or in the past.

H_LHourAreRelative (Read/Write) [String]

Whether or not the H-Hour and the L-Hour have a static time delta. With this option on TRUE, the H-hour and L-hour will always have the same time difference and will adjust automatically when changing either the H-Hour or the L-Hour

Unit Lua Wrapper

AllowMultiMission (Read/Write) [Bool]

Is the mission allowed to have multiple missions assigned to it and will it be considered by the operation planner logic ?

AssignedMissionsQueue (Read/Write) [NLua.LuaTable]

The list of missions assigned to this unit. This is the missions the operation planner evaluate every tick.

8.0 Databases and Templates

8.1 Scenarios vs. Databases

COMMAND has two databases that store all information on platforms (aircraft, ships, submarines, ground facilities and satellites) and components (weapons, sensors, mounts, magazines, communications gear, etc.) used in the simulator. The CWDB database covers the years from 1950-1979, and the DB3K database covers 1980 to the near future.

The databases contain a myriad of platforms. Platforms that span 1980 – those that exist in both databases – are automatically synchronized to ensure they are identical.

The two databases come in different versions. New versions contain additional platforms and components, or fixes to the existing ones. Scenarios store information about what database version they were built with, and when a scenario loads it automatically loads the correct database. This prevents errors caused by scenario/database mismatches, and ensures that the scenario will play exactly the way the scenario author intended.

8.2 Scenario Maintenance

When a scenario is created and saved it is linked to the database it was created with. Some information from the database is saved with the scenario, which in effect locks a scenario to a certain database version. Therefore, the simulator must always load the correct database for the scenarios.

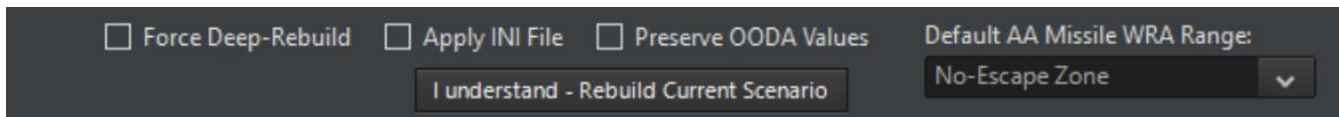
When a platform or component in the database is updated and you want a scenario to reflect those changes, you must migrate (aka rebuild) the scenario to the latest database. Typically when a scenario is under construction and you want to use a platform that was recently added to the database. Or when the latest database contains fixes made to existing platforms or components that you would like to take effect in the scenario.

Rebuilding a scenario with the latest database has to be done in the scenario editor. In addition to rebuilding single scenarios it is also possible to rebuild multiple scenarios in one operation. Single scenarios can be either shallow rebuilt or deep rebuilt. Multi-rebuild of scenarios uses the deep rebuild method.

To open the Scenario Migration window, first load the desired scenario in the Scenario Editor. Then go to 'Editor' => 'Upgrade scenario to latest DB version'.

Note the options in check boxes in the scenario migration dialog box.

- Force Deep rebuild – A deep rebuild will re-initialize all platforms in the scenario to their pristine state, fresh from the database. (see section 8.5)
- Apply INI file – A file that holds the delta of the components of a scenario with an XXX.ini format. (see section 8.7)
- Preserve OODA – Will preserve the current scenarios OODA cycle settings instead of any new OODA cycles in the upgraded database (see section 6.3.4 for OODA information)
- Default AA Missile WRA Range – Allows the designer to set a default WRA range for SAM and AAM.



8.3 Rebuild Single Scenario

To rebuild the currently loaded scenario press the Rebuild Current Scenario button. The lower half of the window will display log messages from the rebuild process. When the Apply INI File option has been selected you will be asked to select the scenario config file (INI file, described later) that shall be used during the rebuild.

8.4 Rebuild Multiple Scenarios

To rebuild multiple scenarios in one operation press the Deep-Rebuild All Scenarios In List button. Follow the instructions in the lower half of the window, including selecting the location of the Scenario List file. This file contains a list of the scenarios to rebuild along with their associated scenario config files (INI files, described later). The Scenario List file has to be generated manually and typically looks like this:

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<ScenarioList>
```

```
  <Scenario>
```

```
    <ScenarioFilePath> <!-- C:\Command\Scenarios\Battle of Chumonchin Chan.scen -->
```

```
  </ScenarioFilePath>
```

```
    <ConfigFilePath> <!-- C:\Command\Scenarios\Battle of Chumonchin Chan.ini --> </ConfigFilePath>
```

```
  </Scenario>
```

```
  <Scenario>
```

```
    <ScenarioFilePath> <!-- C:\Command\Scenarios\Battle of Latakia.scen --> </ScenarioFilePath>
```

```
    <ConfigFilePath> <!-- C:\Command\Scenarios\Battle of Latakia.ini --> </ConfigFilePath>
```

```
  </Scenario>
```

```
</ScenarioList>
```

To add a new scenario to the file, simply copy-paste the lines (tags) for an existing scenario, update the scenario file paths and names, and save the file.

8.5 Shallow Rebuild vs. Deep Rebuild

A shallow scenario rebuild will retain the current inventory of mounts/sensors/stores etc for existing platforms in the scenario. But the performance characteristics of their equipment will be updated as per the latest database version. New units added to the scenario will be inserted as defined in the latest database version.

A deep rebuild will re-initialize all platforms in the scenario to their pristine state, fresh from the database. It resets the properties of each platform and all of their components, including the platform's weapons, sensors etc. It also completely resets damage status, fuel status etc.

Most scenarios can be rebuilt using shallow rebuild however if there is a complex change to the database you must use the deep rebuild function. If you're not sure whether to use shallow or deep rebuild in a scenario, select deep rebuild.

8.6 Log files

The scenario migration (rebuilder) tool will generate two log files located in the \Command\Logs directory. The first file is named SBR log file.txt and records any errors or warning encountered during the rebuild. The second file is named SBR platform list.txt and creates a list of all the platforms present in the rebuilt scenario(s). The platform list also includes aircraft on the ground and docked vessels, and displays the loadouts used by all aircraft. Each platform's scenario name and database name (the two are usually different!), database ID# and scenario GUID is listed as well.

All detected errors are marked ERROR in the log. To look for error messages in the log simply open the log file in Notepad, press F3 or Ctrl + F, and search for the term ERROR.

8.7 Scenario Config Files (INI file)

If any changes/customizations have been made to a platform in a scenario and you would like to carry these over to the new version of the scenario after a deep rebuild, the changes will need to be re-applied. Re-applying the changes can either be done manually in the scenario editor by loading and unloading weapons etc, or automatically through the use of the scenario config file (or INI file). Using a config file saves you having to manually edit the platforms each time.

The scenario config files can either be crafted manually, or exported from an existing scenario. An exported config file will then contain all custom modifications made to the platforms in a scenario before the rebuild, and these can then be applied to the re-initialized platforms in the rebuilt scenario.

The most commonly used functionality is adding and removal of weapons and altering weapon quantities for mounts, mount magazines and magazines. It is also possible to add and remove mounts, magazines, sensors and comm gear (including weapon datalinks).

The scenario config files can edit platforms in the following ways:

- Change mount's weapon quantity (for existing weapons)
- Change mount magazine's weapon qty (for existing weapons)
- Change magazine's weapon qty (for existing weapons).
- Delete weapon on mount
- Delete weapon in mount magazine
- Delete weapon in magazine
- Add new weapon record to mount (Warning: make sure to add associated weapon directors/illuminators or datalinks)
- Add new weapon record to mount magazine, i.e. a mount's local magazine. (Warning: make sure to add associated weapon records on the mount itself)
- Add new weapon record to magazine (aircraft ordnance in ammo dumps and carrier magazines especially)
- Add mount
- Add magazine
- Add sensor
- Add comm gear (typically weapon datalinks or command guidance systems)
- Delete mount
- Delete magazine
- Delete sensor
- Delete comm gear

8.8 Scenario Config File Templates and Delta Templates

Scenario config files (INI files) can either be written manually in a text editor or created automatically by the simulator. Two types of config files can be created by the simulator, templates, and delta templates. Template files contain a simple list of all units present in a scenario, while Delta Template files contain all manually altered mounts, magazines, sensors or comms.

Template: exports the platforms present in the scenario to file. The generated file lists the platforms and their unique GUIDs, and is a starting point for manually crafting a scenario config file. To create the template file, go to Editor -> SBR -> Generate Template. Template files are saved in XML format and will typically look like this:

```
<?xml version="1.0" encoding="utf-8"?>
<ScenarioUnits>
    <Unit_a25be2bd-d06f-4162-959b-b3d7ccc1e5ff>
        <!--LCS 1 Freedom (LCS 1 Freedom [1839])-->
    </Unit_a25be2bd-d06f-4162-959b-b3d7ccc1e5ff>
</ScenarioUnits>
```

Delta Template: exports the delta (difference) between platform configurations in the scenario and those in the database. The Delta Template functionality is most commonly used to export aviation magazine changes or additions. But the generated file will also list platforms that have new and deleted weapons, weapon quantity changes, added/removed mounts and magazines, added and removed sensors, and added and removed communication gear and weapon datalinks. The Delta Template functionality will not export damage or changes in fuel levels.

Example: An aviation magazine has been given a new 0/10000 Mk84 LDGP weapon record which has been loaded with 200 weapons. The Delta Template will attempt to locate the 200/10000 Mk84 LDGP record in the database. Since no exact match is found for this weapon record, the Delta Template will look for a best match using various conditions. It will end up adding a WeaponRecADD and WeaponRecEDIT entry to the scenario config file to first add the 0/10000 weapon record and then load 200 weapons in order to produce the desired result.

The Delta Template will also check mount and sensor arcs to look for differences between the scenario and database. Say a CWIS mount has been deleted and another added with a different arc. This will result in one insert and one delete statement in the Delta Template file.

To create the delta template file, go to Editor -> SBR -> Generate Delta Template. A delta template file would typically look like this:

```
<?xml version="1.0" encoding="utf-8"?>
<ScenarioUnits>
    <Unit_0c4328f6-8900-49c9-a1ee-bbf8e0a2ca92>
        <!--USS Ranger (CV 61 Ranger [Forrestal Class] [1628])-->
        <Mag_2_854>
            <WeaponRemove_1841 />
            <!--AGM-78D Standard [ARM]-->
            <WeaponRemove_150 />
            <!--AIM-7F Sparrow III-->
            <WeaponEdit_1084_8 />
            <!--AN/ALQ-99G [ICAP-2 Block 82] ECM Pod-->
            <WeaponEdit_444_48 />
            <!--AIM-54A Phoenix-->
```

```

    </Mag_2_854>
    </Unit_0c4328f6-8900-49c9-a1ee-bbf8e0a2ca92>
</ScenarioUnits>

```

Note that the template file has comments to ease manual edit of the file later on. The comments contain information such as mount and weapons names, and #IDs from the database. Any errors encountered during the Delta Template file creation will be logged to the \Logs\SBR INI template log file.txt file.

8.9 Editing Scenario Config Files

To edit a scenario config file, open it in Notepad. Config files are stored in XML format and contain an entry for relevant platforms in the scenario. Scenario config files created using the Delta Template functionality will only contain the platforms that differ from the database, and if no differences are found the file will be empty. Scenario config files created using the Template functionality will contain all platforms.

A template file for a scenario with a single ship in the LCS 1 Freedom class will typically look like this:

```

<?xml version="1.0" encoding="utf-8"?>
<ScenarioUnits>
    <Unit_a25be2bd-d06f-4162-959b-b3d7ccc1e5ff>
        <!--LCS 1 Freedom (LCS 1 Freedom [1839])-->
    </Unit_a25be2bd-d06f-4162-959b-b3d7ccc1e5ff>
</ScenarioUnits>

```

Note that 'Unit_a25be2bd-d06f-4162-959b-b3d7ccc1e5ff' is the unique GUID identifier for the LCS 1 in the scenario. It will be different for all individual platforms. The line '<!--LCS 1 Freedom (LCS 1 Freedom [1839])-->' is a comment and can be altered as needed. It is good practice to always have meaningful comments for all lines in the scenario config files.

To add a new torpedo magazine with database ID# 240 to the ship, insert an 'Add Magazine' line. The line also has a comment for future reference:

```

    <MagAdd_240 /><!--Torpedo Magazine [Internal Tubes] -- USN, 2005-->

```

The edited file looks like this:

```

<?xml version="1.0" encoding="utf-8"?>
<ScenarioUnits>
    <Unit_a25be2bd-d06f-4162-959b-b3d7ccc1e5ff>
        <!--LCS 1 Freedom (LCS 1 Freedom [1839])-->
        <MagAdd_240 /><!-- Torpedo Magazine Internal Tubes, USN 2005 -->
    </Unit_a25be2bd-d06f-4162-959b-b3d7ccc1e5ff>
</ScenarioUnits>

```

To apply the changes to the scenario, make sure that the scenario is loaded in the Scenario Editor. Then go to Editor -> SBR -> Apply SBR and select the desired Scenario Config File. Return to the scenario, select the platform, open the Magazines window, and verify that the magazine has been added.

8.10 Scenario Config File Overview

The commands that can be used in the Scenario Config Files are as follows:

Add mount: Mount with database ID# 434 is used as an example below. Note that Coverage has to be added as well, either each segment of a limited arc (Alternative 1), or a full 360 deg arc (Alternative 2). Both have been included in the example however only one should be used in the actual file:

```
<MountAdd_434>
<!--CADS-N-1 [3M87 Kortik]-->
<Cov>

<Seg>PMA1,PMA2,PMF1,PMF2,PS1,PS2,PB1,PB2,SMA1,SMA2,SMF1,SMF2,SS1,SS2,SB1,SB2,</S
eg>
<Seg>360</Seg>
</Cov>
</MountAdd_434>
```

Add magazine: Magazine with database ID# 240 is used as an example below:

```
<MagAdd_240 /><!-- Torpedo Magazine Internal Tubes, USN 2005 -->
```

Add sensor: Sensor with database ID# 2003 is used as an example. Note that Coverage has to be added for both search coverage and illumination/target tracking. The two coverages have been given different arcs for illustration purposes:

```
<SensorAdd_2003>
<!--AN/AWG-9-->
<Cov>
<Seg>360</Seg>
</Cov>
<Cov_Ill>
<Seg>PMA1,PMA2,PMF1,PMF2,PS1,PS2,</Seg>
</Cov_Ill>
</SensorAdd_4>
```

Add comm link: Missile datalink with database ID# 3 used as an example:

```
<CommAdd_3 /><!--SA-10 Missile Datalink-->
```

Delete mount, magazine, sensor or comm link: Platforms carry a variety of equipment (components), and in many cases there are several mounts, sensors, etc, of the same type but with different coverage (arcs). In order to precisely specify what component to delete, you have to specify not only the database ID# of the component but also the component's number in the ship component list. In the example below we're deleting component #3 with database ID# 222 from a platform:

```
<MountRemove_3_222 />
<MagRemove_3_222 />
<SensorRemove_3_222 />
<CommRemove_3_222 />
```

If component #3 with ID# 222 is not found on the current platform a popup message will inform you of this fact, and an error message will be saved in the log file.

Edit, add and remove weapon record on mount, in mount magazines or in magazines: The below example edits Sea Sparrow mount with component ID #3 and database ID# 552. Pre-defined weapon records in the database are added to the mount and its on-board mount magazine (WeaponRecAdd). The weapon quantity is then altered (WeaponEdit), and some of the weapons are then removed (WeaponRemove):

```

<Mount_3_552><!--Mk29 Sea Sparrow-->
<WeaponRecAdd_5188 /><!--RIM-8J Talos-->
<WeaponRecAdd_1557 /><!--RUM-139A VLA [Mk46 Mod 5]-->
<WeaponRecAdd_2626 /><!--Roland 3-->
  <WeaponRecAdd_944 /><!--RUR-5A Mod 3 ASROC RTD [10kT Nuclear DC]-->
<WeaponEdit_1687_2 /><!--RIM-7M Sea Sparrow-->
<WeaponEdit_885_22 /><!--RIM-8J Talos-->
<WeaponEdit_1643_77 /><!--RUR-5A Mod 3 ASROC RTD [10kT Nuclear DC]-->
<WeaponRemove_233 /><!--Mk182 SRBOC Chaff [Seduction]-->
<MountMag>
<WeaponRecAdd_5188 /><!--RIM-8J Talos-->
<WeaponRecAdd_154 /><!--RUM-139C VLA [Mk54]-->
<WeaponRecAdd_4015 /><!--S-8KO 80mm Rocket-->
<WeaponRecAdd_2765 /><!--SAM-3 Kin-SAM [Type 93]-->
<WeaponEdit_885_3 /><!--RIM-8J Talos-->
<WeaponEdit_1193_88 /><!--SAM-3 Kin-SAM [Type 93]-->
<WeaponRemove_233 /><!--Mk182 SRBOC Chaff [Seduction]-->
</MountMag>
</Mount_3_552>

```

The following example edits the SRBOC magazine with component ID# 1 and database ID#717. Several pre-defined weapon records in the database are added (WeaponRecAdd), the number of weapons is then altered (WeaponEdit), and some of the weapons are removed (WeaponRemove):

```

<Mag_1_717><!--Mk36 SRBOC-->
<WeaponRecAdd_2541 /><!--Generic Chaff Salvo [5x Cartridges]-->
<WeaponRecAdd_5188 /><!--RIM-8J Talos-->
<WeaponRecAdd_154 /><!--RUM-139C VLA [Mk54]-->
<WeaponRecAdd_4015 /><!--S-8KO 80mm Rocket-->
<WeaponRecAdd_2765 /><!--SAM-3 Kin-SAM [Type 93]-->
<WeaponEdit_232_222 /><!--Mk182 SRBOC Chaff [Seduction]-->
<WeaponEdit_885_3 /><!--RIM-8J Talos-->
<WeaponEdit_1193_88 /><!--SAM-3 Kin-SAM [Type 93]-->
<WeaponRemove_233 /><!--Mk182 SRBOC Chaff [Seduction]-->
</Mag_1_717>

```

If no mount or magazine on the current platform matches the component number and database ID# combination a popup message will inform you of this fact. An error message will also be saved in the log file.

9.0 COMBAT

Combat in Command involves several basic factors. First one must see the opponent. Then you must hit the opponent. The former involves sensors, the latter involves weapons.

9.1 Sensors

COMMAND has several different types of sensors that can be divided into four main categories: radar, ESM, sonar, and optical sensors. All sensors have a physical height associated with each of the components. You often see radars mounted on tall towers or on the tops of the masts of ships, this is done to improve the sensors visibility of the horizon. The mast height of a given unit can be explicitly set by the unit designer, or it will be estimated based on the overall size of the platform.

9.1.1 Radar.

Radar functions by sending radio waves at the target. COMMAND contains radars from the earliest and crudest sets to extremely advanced electronic arrays. As radars are active, they all have an inherent risk of being jammed and giving their whereabouts away. However, a more advanced radar will be less vulnerable to this, especially if faced against a less advanced jammer.

Radars have a limited field of view they can search and track their targets within. Fire control radars in particular often have very small beamwidths of a few degrees that can only concentrate their emissions in a limited sector of the sky. Complicating this equation is how mechanically steered radar beams require a non-zero amount of time to orient themselves to a certain direction. Because of this, long range radars require several seconds to complete a circular sweep and many early aircraft radars had very limited fields of view for search purposes. In COMMAND, this effect is modelled in several ways, and understanding the limitations often creates new tactical opportunities.

Once a mechanically steered radar wishes to hold its beam on a certain target, it can no longer 'turn' to face another target. If the radar is an illumination type radar, it has to have a continuous line of sight on that target at the cost of not being able to engage other targets. In addition, all radars have vertical and horizontal limits based on the generation and type of radar. These limits provide unique options for avoiding the beam of the radar by simply staying out its effective field of view. This also means that approaching a SAM battery, for example, from multiple directions simultaneously prevents the SAM from reliably engaging in all directions. Note that depending on the generation of the radar, the ability to detect targets that are 'below' the radar and are close to ground clutter may be limited.

Compare several American fighter radars:

AN/APQ-120 (F-4): Basic mechanical scan radar. The earliest and least capable. With no "track while scan" function, it can either guide missiles to its painted target *or* search for contacts, but not both at the same time.

AN/APG-63 (F-15A): Mechanical scanning, frequency agile. Frequency agile radars are less susceptible to jamming and "doppler notching", an air combat technique where an aircraft flies perpendicular to the emitter.

AN/APG-70 (F-15C): Mechanical scanning, frequency agile, NCTR-JEM. In addition to move advanced electronics, an F-15C with this radar will have the ability to identify enemy aircraft flying head-on, as its NCTR radar “counts” the fan blades of their engines.

AN/APG-77 (F-22): Active electronically scanned array, NCTR-NBILST. This incredibly advanced radar is electronically scanned, making gimmicks like doppler notching pointless. Its NBILST (Narrow Beam Interlocking Search and Track) NCTR function means it can positively identify aerial targets from any angle. Phased array radars also do not have the issues with scanning up and down that less sophisticated models do.

Electronically scanned arrays have other benefits besides having enhanced detection capabilities. They are better able to filter out jamming thanks to their frequency agility and more importantly, they are more resistant to ground clutter once a target has been initially located. Ground clutter can still affect their search modes as in conventional radar, however in tracking modes, the electronic array radars are very precise, efficient and difficult to force a broken lock. One of the downsides of an electronically scanned array radar is that the strongest detection capability comes from the center of the radar projecting perpendicular to the face of the array. This means that forces wishing to be undetected can avoid being in the center lobe of the radar, and have enjoy a significantly reduced chance of detection when compared to conventional pulse based radar.

9.1.2 ESM

Electronic Support Measures are passive systems that can detect electronic emissions (radar, OECM jammers). They range from simple, crude radar warning receivers that can give a vague and imprecise (yet still very important) notification that an emitter is somewhere nearby to advanced detectors, that, based on the electronic sources, can conclude that the emitting platform is a MiG-29 Fulcrum A with a N-019EB radar. In COMMAND, aircraft will now start defensive maneuvering the moment that they detect illumination by a fire control radar, even before a missile is detected on route to the aircraft.

ESM platforms, even more precise ones, have an inherent lack of precision compared with active sensors that constantly “paint” the target. However, having multiple ESM-equipped units present can counter that to a large extent by triangulating the signals.

Optical. Almost every human has two optical sensors in the front of their head. In addition to the “Mk 1 Eyeballs”, COMMAND boasts many types of *electro*-optical sensors, including cameras and infrared systems. These range from early, cumbersome TV cameras to incredibly sophisticated imaging infrared systems.

The biggest advantage optical systems offer over radars is their passivity. Because optical systems are considered “passive” sensors and do not emit, they are not vulnerable to being picked up by ESM the way radars are. Imaging optical systems also can confirm the identity of a target in a way that most radars cannot.

However, optical systems also have significant disadvantages. One is a “looking through a straw” effect where the camera has a very limited field of view. Another is a lack of precision, with electro-optical systems working by themselves can only gain precise target information at very short ranges.

9.1.3 Sonar

Active sonar is like underwater “radar” (an apt comparison, as both reflect waves of energy off the target). It creates a more precise contact at the expense of being easily detected. Passive sonar has many of the same weaknesses as electro-optical systems (imprecise, especially at long range) but also their strengths (passive,

does not give away the sensor when being used). Vertical flank arrays are now used to create more precise target tracks against underwater targets.

Towed arrays are used on many ships and submarines. They can be held a distance away, making them less vulnerable to their own engine noises, and they can sit below the thermal layer, reaching past it the way a hull sonar on a surface ship cannot.

9.1.4 PCLS

One newer type of electronic detection is the passive coherent location system (PCLS). This system uses the reflection of electronic waves against the body of an object but instead of utilizing a transmitter/receiver combination as in conventional radar, it uses existing electromagnetic noise created by everything from cell phone towers to VOR beacons. The PCLS monitors the propagation and properties of these existing signals with multiple receivers spread apart to determine the presence of an object without having an active signal of their own. In order for this process to work, there has to be an appropriate set of transmitters as well as a constellation of typically ground based receivers.

In COMMAND, PCL systems are modelled by placing several sources of electromagnetic emission such as radio and tv towers and the setting up a PCL platform in a position where it can receive both the 'echoes' of the target as well as the source of the radiating signal. It typically takes three or more accurate sources and reflection to develop a strong enough track and while PCL can determine the vicinity of the target, a targeting quality track will require alternate detection systems to acquire that target. PCL is therefore an excellent and passive early warning system, but upon detection, traditional sensors are required for a quality track.

9.1.5 Other (MAD)

Magnetic Anomaly Detectors are used by anti-submarine aircraft. They search by measuring magnetism and looking for large anomalous deposits of metal (ie, submarines) under the water. Thus, the ideal MAD subject is a large submarine (lots of that metal!) in deep water (less background metal to get in the way)

While passive, MADs are ***not precision sensors***. They do not have a range component and only have a vague bearing element-this is why maritime patrol aircraft still carry all the sonobuoys that they do.

9.2 Weapons

9.2.1 Unguided Weapons

Unguided weapons include guns, bombs, and rockets. Their accuracy is determinant on many factors, including the skill of the operator, the type of fire control system, and the range to the target.

Ballistic missiles represent one class of unguided weapons, and have unique properties in the way that they are deployed, travel and operate. Most ballistic missiles operate by having a large rocket power a payload into a tall parabolic arc. When the rocket burns out, the weapon continues to glide to its apogee before gravity and drag catch up and the weapon begins to glide back towards the ground. In some ballistic missiles, particularly ICBMs, the payload of the weapon detaches from the main body of the weapon and as such presents a much smaller and more difficult target to destroy. In addition to the payload containing a warhead, ballistic missiles

may now be equipped with decoys or hypersonic glide vehicles (HGVs) to increase the probability of striking the target and not being intercepted.

COMMAND models all phases of a ballistic missile trajectory in detail and the challenge in detecting and successfully intercepting these weapons. Although many countries throughout the world have developed dedicated systems to defeat these weapons, the incredibly high speeds and the enhanced maneuverability of recent systems has resulted in several countries seeking systems other than guided missiles, including sci-fi like lasers and high-velocity weapons. COMMAND models all phases of ABM carefully and dedicated systems for the purposes of shooting down ballistic missiles fire when the proper opportunity is detected.

9.2.2 Guided Weapons

Guided weapons include guided missiles and so-called “smart bombs”. The guidance systems present in COMMAND include:

- Inertial guidance. The weapon is programmed to fire on a specific set of coordinates and its internal systems direct it (to the best of their ability) to those coordinates. Weapons with inertial guidance range from early ballistic missiles that will be lucky to land in the boundaries of their target city to advanced, ultra-precise weapons.
- Optical guidance. The weapon homes in on an image of the target. Optical systems range from early, crude ones like the GBU-8 HOBOS to advanced imaging infrared systems (that can even act as makeshift sensors in their own right).
- Infrared Guidance. The weapon homes in on the heat signature of the target. In modern seekers with imaging systems this overlaps with “optical guidance”, but it applies also to early heat-seeking missiles. Anti-air IR missiles come in three varieties:
 - Stern chase (need to be right behind to get a good lock, ie AIM-9b/AA-2),
 - Rear Aspect (still need to be behind yet are more flexible than stern-chase weapons, IE, AIM-9H/AA-8)
 - All Aspect (can fire on the target from any angle, ie AIM-9L+/AA-11).
- Semi-Active Radar Guidance: In short, this guidance system involves the missile homing on a target “painted” by a radar. Breaking the lock (such as forcing the launcher to turn away) will cause it to lose control.
- Semi-Active Laser Guidance: This is the classic laser guided bomb, where the launcher paints the target with a laser designator and the weapon homes in on the designator. Some types of weapon can use “buddy lasing” (where an aircraft or even a ground unit can designate the target while the other drops from farther away)
- Active Radar Homing: This involves an emitting radar system on the weapon itself. ARH was pioneered on anti-ship missiles like the SS-N-2 that were big and only had to engage slow targets. Later it became miniaturized enough to be used for AAW weapons like the AMRAAM.

It’s important to note that ARH does not necessarily mean the weapon can just be fired “blindly”. Most AMRAAM-style ARH air-to-air weapons (except for newer ones with an appropriate datalink and a friendly illuminator nearby, like an E-2D Advanced Hawkeye and AIM-120D advanced AMRAAM) need the launching platform to detect and illuminate the target with its own radar before launching.

Guided missiles have an additional challenge to employment, and that is in how they propelled towards the target. While the popular view of missiles in movies depicts a slow-moving projectile powered continuously

throughout flight, only cruise missiles have that type of capability. The majority of missiles use a boost-coast model for propulsion meaning that they burn their motors for a certain amount of time to build up speed, and then glide towards the target using the energy they accumulated during their initial acceleration. Not only does this mean that a typical guided weapon operates at its peak speed for only a few moments before atmospheric drag and gravity takes over, slowing the weapon down. This tendency for a missile to exhaust itself in thicker air along with losing energy as it maneuvers is one of the defenses a modern pilot has against these formidable weapons.

It is also worth noting that some guided weapons, particularly those optimized for very long-range engagements, employ not only a boost-glide model, but utilize a very high lofted trajectory whereby the guided weapon is shot to very high altitudes where atmospheric drag is less and then glides downwards onto the target constantly accelerating due to the force of gravity. Many of these types of guided weapons are 'smart' in that they will select the optimum trajectory depending on the interception time and range to the target.

There are some guided weapons modelled in command such as the RIM-174 (SM-6) that use a continuous burning motor. These types of weapons are usually employed against ballistic missiles or very high-altitude targets and will also include maneuvering jets to improve interception accuracy in exoatmospheric engagements.

COMMAND also models loiter-type weapons such as the ALARM anti-radiation missile. This weapon is designed in such a way that if it loses contact with an emitting target, it deploys a parachute and begins gliding downwards. When the target radar system is reacquired, the parachute is jettisoned and the rocket motor fires again providing the required thrust to complete the weapon's journey towards the target.

Some guided weapons have multiple types of guidance built into them. This provides an additional layer of redundancy should one of the seekers fail or the target is impossible to detect using it. An example of one of these weapons is the AIM-88E AARGM, a specialized radar-destroying missile that can use passive radar guidance to home into a radar but in the event that the radar is turned off, it can use a combination of GPS and active radar to guide towards the target.

9.2.3 Torpedoes

Torpedoes swim through the water. There are two main guidance systems in the initial torpedo phase:

- Inertially guided torpedoes are the earliest types of torpedoes in COMMAND. An example is the classic Mk14 World War II torpedo. They have no way of changing course once fired.

- Wire guided torpedoes are connected to their parent submarines and can be steered as if they were units in their own right via the Plot Course function while the wire is active (one use of this is to prevent the torpedoes from piling up on one target).

However, turning extremely fast (as in, "the sub's life is in danger" fast) or just moving quickly (over ten knots) while firing the torpedoes will likely cause the wires to break. A torpedo with a broken guidance wire goes autonomous, activating its terminal seeker and going out of the launcher's control.

- Torpedo terminal guidance includes traditional sonar guidance and wake-homing. The former is considerably easier to decoy.

9.2.4 Lasers

-Since the first pulp hero obliterated the first space monster with a ray gun, people have dreamed of turning fiction into fact. Now, after decades of development and the successful use of military lasers for targeting purposes, the dream is finally becoming a reality. Lasers are modeled in COMMAND.

-Lasers come in four categories, each with their own characteristics. Chemical Oxygen Iodine (COIL), Deuterium Fluoride, Carbon Dioxide, and Solid State/Fiber. COIL and DF lasers use chemical feeds as “ammunition” and are thus exhausted and reloaded like conventional weapons, while CO2 and SS/F lasers draw on their parent’s power plants and can keep firing as long as it lasts.

-As they move at the speed of light, lasers attack their target instantly, giving it no evasion bonus. However, they can only immediately destroy the smallest targets, requiring sustained exposure for larger ones. Turning on the “Aircraft Damage” setting is thus highly recommended for scenarios that involve lasers.

HVPs/railguns:

The High Velocity Projectiles currently in COMMAND are modeled similarly to missiles, only fired from certain guns. They are treated as “guided weapons” in the database and offer their users plentiful, long-range, accurate strike capability.

9.2.5 Electromagnetic Pulse (EMP)

Electro-magnetic pulses serve to destroy electronics. The effects they have on their targets depend on the intensity of the pulse (which is *not* affected by the warhead’s yield in the classic “nuclear EMP” scenario but is affected strongly by its detonation altitude), the operating status of the target (active sensors are more vulnerable) and the technology generation (older -think vacuum tube-) electronics are the most resistant, while increasingly advanced ones become increasingly more susceptible).

EMPs can be unleashed in multiple different ways.

- As the byproduct of a nuclear warhead used for other purposes.
- By a Lua-induced “Add Explosion” action, particularly detonated at an extremely high altitude for maximum effect.
- By a “one-click EMP” attack. Take a suitable weapon, manually allocate it, and in the upper right “weapons allocated window”, right click on the weapon record and select “Special high altitude detonation”. It will then detonate at the optimum altitude for an EMP wave.
- By a weapon with a non-nuclear “tactical EMP” warhead.

EMP warheads come in two main types: directed and omnidirectional. Omnidirectional or O-EMP can be compared to a hand grenade in how once it detonates, it creates a blast that affects all electronics in spherical radius emanating from the blast point. Directional or D-EMP weapons are designed to focus the effects of the EMP burst on a small target. O-EMP is effective at causing less damage but to more targets whereas D-EMP is better at disabling a specific system through concentration of the effect.

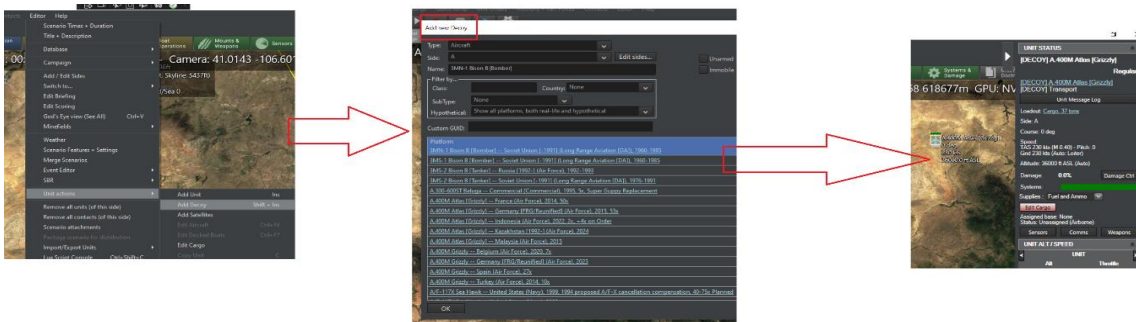
9.2.6 Warhead Types

Warheads can come in various types, from conventional explosives to area ones like airbursts, napalm and clusters (little penetration but a “smothering” effect and lots of flames/secondary damage) to specially designed penetrators (those that can “punch above their weight” by breaking through deeply). As in real life, damage in COMMAND can manifest in strange ways.

9.2.7 Decoys

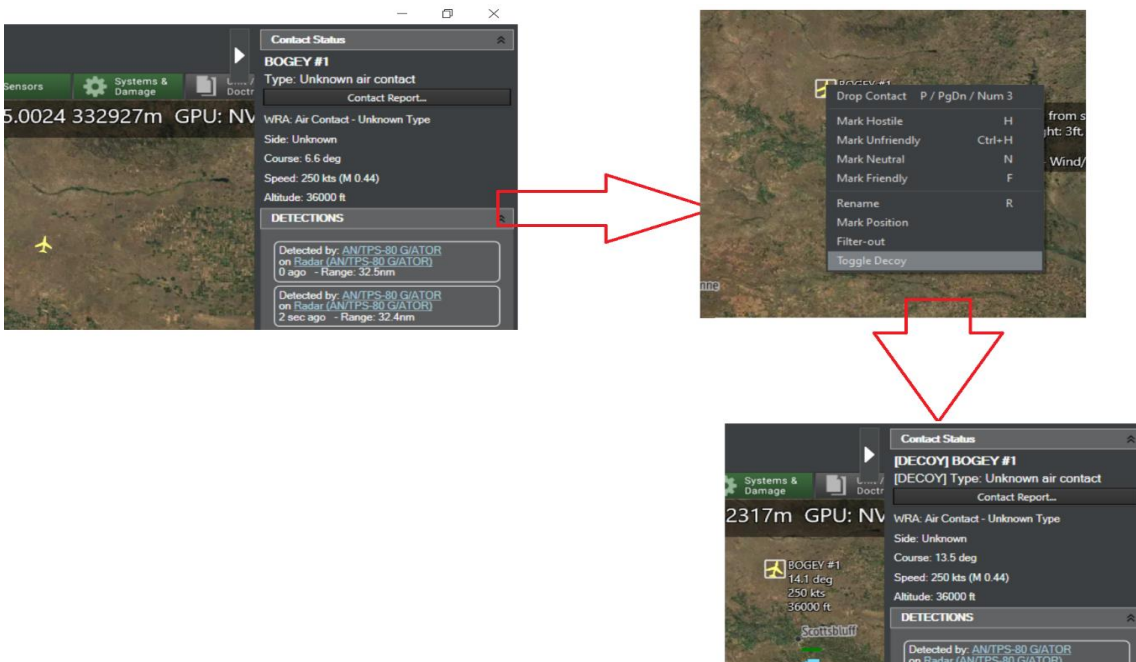
Decoys are units completely identical to the “real” ones but with a Decoy flag set to true. In the real world a decoy can move or not, be armed or not. In order to achieve maximum flexibility a decoy is created as an exact copy of the mimicking unit with the only exception of a flag that indicates this as a decoy.

The scenario editor can freely customize the unit removing weapons or propulsions until it reaches the desired level of simulation.



The side that own the unit will always be able to see that the unit is a decoy thanks to the tag in the right column.

Decoy identification:



The simulation will make no effort in auto-identify a decoy. It will be the player that once identified a unit as a decoy can right click on it and mark it as decoy. The specific defending unit holds a list of side ID that have been marked it as a decoy. Once the unit is marked as a decoy (independently of the fact that the unit is indeed a decoy) the unit is always treated as a waste of ammo and no attempt at auto engage is made, although manual fire will still be possible.

The unit marked as decoy will change its icons with the decoy icon already in the symbol folder and will have the Decoy tag in the right column when inspected. A unit can be unmarked as decoy by right clicking on it and select again “toggle decoy”. This will revert the unit to be considered as a real unit of the identified type

9.2 Battle

In the combat itself, there are several “steps” gone through before the combat is resolved. The first step is identification. Targets (for attackers) and threats (for defenders) have to be resolved in a process that takes time depending on the unit’s proficiency and OODA clock (see 6.3.4, Database Viewer). The next step is actually engaging, where the unit moves in range and fires its weapons. This may sound simple, but in execution, it’s anything but.

9.2.1 Air Combat

An air unit’s ability to engage is dependent on more than just pure range. It also depends on sensors and/or the capability of the weapon. So at one extreme, there are early stern-chase infrared missiles that require the attacker to line up straight behind the target and early SARH missiles that render the attacker’s radar unable to do anything else while it guides them. At the other are all-aspect, off-boresight missiles able to be launched from incredible positions with a helmet mounted sight and CEC-datalinked active radar missiles able to be launched from far away and guided by another aircraft with the launcher never needing to turn on its position-exposing radar.

The target may not see anything and only hope the aiming of the guns is poor or the missile prone to inaccuracy (This is historically accurate, as the overwhelming majority of losers in air combat never saw anything until it was too late). Should they notice, the target shifts to “engaged defensive” and attempts countermeasures. These countermeasures include dodging, countermeasures, and aggressive maneuvering.

Dodging, trying to “beam” (fly perpendicular to) the weapon or radar. This results in a dice roll that gives it a penalty to hit. The penalty is highly dependent on the target’s skill and aircraft agility. Even a master pilot can’t do much in a lumbering cargo plane or heavy bomber, but in a nimble fighter, a novice can lose a dogfight to a plane of an earlier period. Dodging does not apply just to missiles. The enemy aircraft itself or the radar can trigger evasion, the latter to attempt “doppler-notching”, a way of exploiting limitations of (particularly earlier) radars. Aircraft maneuvering in dogfights is affected by physics, which range from airframe limitations (aircraft have to roll into turns) to pilot limitations (as seen on the right-hand display under G-force).

Deploying countermeasures, such as chaff and DECM against radar guided missiles or flares/IRCM against infrared ones. This is highly dependent on the technology level of both weapon and countermeasure, and is treated as a separate calculation before the dodge phase.

Example: A shah-vintage Iranian F-4 Phantom engages a modern, highly skilled American fighter over the Strait of Hormuz with ancient early AIM-7s. While a victory is still possible, it's a steep hill to climb. Even before the skill of the American pilot and the weaker kinematics of the Sparrow come into effect, the missile must overcome a chaff release and electronic jammer, both of which it is highly susceptible to. When the situation is reversed, the old electronics and chaff of a Phantom or even Tomcat offer far less in the way of protection against an AMRAAM. If the target survives or notices the attack before it hits and has an appropriate weapon, they may be able to maneuver and fire back. Then the situation repeats itself.

Other aircraft are not the only danger that modern aircraft face, and thanks to improved missile and radar designs, surface-to-air missile batteries have evolved significantly and their efficiency and effectiveness against flying targets cannot be understated. These systems are able to launch missiles that routinely travel at several times the speed of sound with enough accuracy to strike ballistic missiles from hundreds of miles away.

Once an aircraft is fired upon by a SAM, it has several options at its disposal in order to reduce the probability of being struck by the weapon. The first maneuver it to simply turn the aircraft around and run at maximum speed away from the weapon. Since most SAMs use a boost-glide method to get to their target, this means that although the weapon is moving quite fast compared to the aircraft a few moments after launch, atmospheric drag and maneuvering can significantly slow the missile down before it covers the distance to its target. This means that if the aircraft can build up enough speed, it can literally outrun the missile. Since missiles can only travel so slow due before stalling or their guidance systems can only run so long before their batteries die, the weapon can be rendered ineffective.

If a SAM is fired at a close enough range where it has enough energy to still catch up, the other option a pilot has would be to try to defeat the weapon by putting something between itself and guiding radar. This strategy works if the terrain allows it, the aircraft has enough to get to the proper altitude and if the missile launched from SAM battery does not have on-board guidance that can hold onto the target.

The final option, also simulated in COMMAND is attempting to kinematically defeat the missile through aggressive maneuvering close to the time of impact. It can be possible through beaming or a high-g maneuvers to create a condition where the missile does not have enough energy or guidance to match the sudden maneuver. This is often a last-ditch maneuver when other means of defeating the incoming missile is not possible. COMMAND simulates these techniques based on the range of the missile to the target aircraft and the available energy.

When aircraft are used in formations, many different options control the behavior of the wingmen when responding to contacts. AI wingmen can be told to engage and investigate unknown targets and in order to keep them 'in line', COMMAND calculates all distances based on the formation LEAD as opposed to the entire group.

9.2.2 Naval Combat

With naval guns, accuracy is highly dependent on the type of fire control setting, the range to the target, and the stability of the gun platform. A combination of calm seas and a heavy ship does wonders for accuracy, while a light ship bouncing in rough waters is the opposite. They may both be firing naval guns that are very similar or even exactly the same, but a small patrol vessel with manual control alone in very wavy seas will not hit as well as a large cruiser or battleship in a calm ocean with intact, advanced fire control systems.

With missiles, the target has more of a chance to defend itself. The first line of defence is to try and shoot the missile down. This is a task dependent on the quality of both the launcher and target. Some weapons systems may be unable to engage low-flying or fast enough missiles at all, while others will struggle and have a low probability of success. The second is to deploy countermeasures (chaff/flares and DECM), which work similarly to those on aircraft.

The OODA clock of the target, affected by its technological sophistication and proficiency, also plays a gigantic role. The OODA clock and speed of the attacking missile determine how many shots the defender can fire before the missile hits-or if they even can fight back at all. A old and/or slow system may be completely helpless against an oncoming threat. No one said naval combat was fair.

(In real life, since anti-ship missiles were invented, there has been a debate about 'sneakers' vs 'streakers', whether it's more effective to have a missile rely on stealth or speed to overwhelm the opponent's OODA cycle. Command gives you the opportunity to test it out in all sorts of conditions.)

9.2.3 Submarine Combat

Submarines are similar to surface ships in many ways, and outright identical if they're old-style "undersea boats" engaging on the surface with deck guns. But they're also very different in two important areas.

The first is that submarines are both stealthier than nearly all surface ships and have worse "vision". This largely applies even in extreme cases. Noisy first-generation nuclear submarines still have characteristics no surface ship can match, and modern ones with ultra-advanced sonar still can't see nearly as far as a networked surface combatant. The second is that, surrounded by crushing water, submarines are incredibly fragile. There's a reason why submarine crews ranked at the top of the proportionate casualty lists in the World Wars.

Thus, it's important not to get detected if possible. Use of the thermal layer (see below) is important. So is the realization that a submarine can be inferred and moved against even if it cannot be detected. Both inbound torpedoes ("torpedo datum") and ships suddenly suffering an undetected torpedo hit ("flaming datum") will cause a submarine contact to be generated for the "victim" side. Likewise, a submarine-launched missile observed leaving the water will create a contact next to the spot where the launch was detected.

Submarines come in diesel and nuclear varieties. Although most of the gap has been closed in present times, earlier nuclear submarines are incredibly noisy compared to their diesel counterparts. Furthermore, their reactors and pumps mean that they remain loud even when stationary or moving very slowly.

However, what the nuclear submarines lack in stealth, they more than make up for in terms of speed. Not only can most of them move faster than comparable diesel submarines, but all of them can move *consistently* faster. Whereas as a submerged diesel running at full blast will burn through its batteries extremely quickly, a nuclear submarine can keep going and going like a certain battery mascot.

Moving quickly is an effective strategy against submarines, especially diesels. However, not all HVUs in an area may be able to move fast enough to follow that strategy.

Submarine sonars get a clear picture only with a “direct-path” contact, a fairly close area to the sound, roughly analogous to the visual horizon. In deep water, a submarine with a powerful sonar may have a series of green rings around it. These represent the convergence zones. Units entering the convergence zones can be detected by the sonar, although their uncertainty zone is far wider than it would be under a direct-path contact.

In shallow water, convergence zone contacts are not possible. This is a known limitation of submarines in the littorals, and the other side of the effects that also make them deadly.

Understanding Depth Bands And The Thermal Layer

What follows is a brief summary of how the different depth ranges (reflected on the depth-presets on the unit throttle/altitude window) are modeled in COMMAND, and the tactical implications of each.

Towed arrays and/or VDS systems are represented by the red tails on each platform.

Periscope depth: This is the only depth at which subs can use their above-water sensors (periscope, ESM, radar etc.). A strong surface duct is usually present, which magnifies the transmission range of noise well beyond nominal (in fact, in calm seas a surface ship can use this duct to get early warning of threats that are visually/electronically stealthy but acoustically very noisy, such as small high-speed boats). Bad weather significantly degrades this duct's effect. All sub-launched missiles can be launched at this depth. Because of the small water pressure cavitation sets in even at relatively low throttle settings. This is the optimum depth for detection against surface ships (and counter-detection by hull sonars on surface ships, as well as "shallow"-set sonobuoys and dipping sonars).

Shallow: Some (but not all) sub-launched missiles can be launched at this depth. The surface duct is still present but significantly weaker. This depth is suitable for keeping an eye (or ear, as it goes) on surface traffic and staying near (or at) missile-launch conditions while keeping out of the strong surface duct (reduce counter-detection by air/surface) and keeping close to the layer below for a quick dive if necessary (ditto).

Just above layer: Completely out of the surface duct, this depth reduces interaction with air/surface matters even more, at the benefit of ASW. Indeed, this is the ideal hunting ground for ASW-oriented subs with towed arrays. This is because subs at this depth automatically trail their arrays *below* the layer; thus they maximize their detection range against anything below while masking themselves with the layer (and retaining their above-layer search ability with their hull sonars). Cavitation speed is significantly higher here. No missiles can be used at this depth or below.

In-layer: Like above, the towed array hangs below the layer but the counter-detection reduction is not as great (sound has to go through a lot less to reach an enemy sensor). Also the unpredictable mixup of warm and cold water at this depth range significantly reduces detection ranges against other subs also in the layer (think Mutara Nebula from Star Trek II).

Just under layer - the Deep Sound Channel: This is by far the most transmission-friendly depth band, greatly magnifying detection ranges. Surface ships and subs just above the layer both trail their VDS or towed arrays in this band. Sonobuoys and dipping sonars in "deep" setting also operate here. As a result, this is the worst place to be if you are hunted by modern ASW forces (or conversely, the best place to be if you want to attract attention). As a consolation, cavitation speed rises even higher. ASW-tasked subs without towed arrays often loiter in this band to maximize the range of their hull sonars. Surface and shallow sonars as well as "shallow"-

set sonobuoys and dipping sonars have their ranges drastically cut against under-layer targets, and in some cases may not be able to get through at all.

The Great Deep: The DSC still has some influence here but not as great. Cavitation sets in only if you go flat-out (modern subs do not cavitate at all here, even at flank). If the sea bottom is shallower than the sub's rated depth, the sub can "fly nap of the earth" or even go belly-up (sit on the bottom) and get the benefit of greatly reduced active sonar echo. This is generally the ideal depth for "transit" mode, when the emphasis is on moving from A to B rather than hunting, or for stationary ambush.

Please note that the layer ceiling and bottom, as well as its "thickness" (and hence absorption rate) are not fixed, but vary according to latitude, local depth and local temperature (the estimated actual values are shown on the map cursor). Trying to manually keep a submarine at the edges of the layer as these values shift can be a very tedious job, hence the depth presets which automate this.

Convergence Zones in Command:

- CZ detections are possible only if the local depth provides at least 600ft/200m clearance under the target.
- CZ information (range estimates) is displayed on the map mouse cursor right alongside the thermal layer info.
- CZ intervals range from 40nm in the poles to 20nm in the equator, depending also on local temperature.
- An arbitrary number of CZs are supported, if the sound is powerful enough and the receiver sensitive enough. Maximum direct-path range is 20k yards (~9.5nm).
- CZ area depth (i.e., ring thickness) is assumed to be 5nm, so the actual detection range at each CZ interval may be plus/minus 2.5nm.

To verify if CZs can indeed form between the sensor and target, we check not only their respective local depths but also the depths at the nadirs of each CZ curve; so for example the sensor and the target may both have an abyss below them but if an underwater ridge high enough is between them it may well block the CZ path.

Sonar operators are assumed to be proficient enough to discriminate between a direct-path contact and a CZ one (by examining both aural tones and the bearing rate); as a result, the generated AOU will either start from the sensor platform and extend to max direct-path range, or (in the case of a CZ detection) start from the innermost CZ interval and extend to max sensor range.

9.2.4 Mine Warfare

Since World War II, naval mines have sunk or damaged more ships than any other weapon. Immense resources have been directed into both creating and neutralizing minefields. COMMAND thus includes an exhaustively modeled mine warfare system.

There are two ways of laying mines in COMMAND. The first is the "prefab" way in the scenario editor. Create a box of reference points, make sure that they're selected, take note of the water's depth in the highlighted area, and then go to "Minefields" in the editor drop down menu. Select "create minefield in designated area" and a menu will appear containing a long list of naval mines in the current database. These range from named specific mines to a list of "generic mines" of various types. The important kinds are:

- Moored mines. The most stereotypical naval mines, these can be laid at a fairly great depth thanks to their moorings keeping the warhead at a dangerous level. However, the moorings also make them easier to spot (and therefore neutralize).
- Bottom mines. These, true to their name, sit at the bottom of the sea. They are lower-profile than moored mines and thus hard to detect, but because they depend on their blast wave reaching the surface, they have to be laid in shallow waters only. Some bottom mines are “mobile”, deployed by “swimming” out of torpedo tubes at a standoff distance before stopping.
- Floating/drifting mines. These stay on the surface and are thus the most visible. Floaters are very difficult for aircraft to take down.
- Rising mines. Arguably the most dangerous type of mine, rising mines sit on the bottom and then release a payload when they detect their target, which can be an unguided rocket or a homing torpedo like the CAPTOR. Rising mines combine the stealth of bottom mines with the deployment depth of moored mines, making them highly dangerous.

Once the type of mine is selected, choose the number to be deployed (if in doubt, add more), and press “ADD”. The editor will deploy as many mines as it can in an irregular pattern. Frequently there will be less than the stated number deployed even if the water is shallow enough.

The second way is via a minelaying mission. See 7.2.5, Mining Missions. Use air or sea units with a mine loadout (for the latter, mine weapons records can be added to existing units if need be) and execute the mission.

In both types of minelaying, the deployment will be uneven and seemingly erratic. This is deliberate to ensure that opponents can’t accurately guess the minefield’s location after detecting a few of them. Especially if it’s a prefab layout, make sure the mine placement is satisfactory. If it isn’t, then removing the minefields via the “remove mines from designated area” function may be necessary.

Once the minefields are placed and units head towards them, calculations begin when the units get close enough. Its detonation/payload release distance will depend on the type of unit and sophistication of the mine. An under-the-keel blast is highly dangerous and will destroy most ships. A distant blast may not be much on its own, but several can still be cumulatively damaging.

To counter mines requires ships and aircraft with the appropriate systems. Countering mines is conducted via mine counter missions (see 7.2.6, Mine-Clearing Missions).

- The two types of mine countermeasures are sweeping and hunting. Sweeping involves deliberately activating the mines via sweeping gear (visible in the DB viewer and on the map when activated) so that they will explode at a less lethal distance. Many types of mines are unsweepable and have to be neutralized one at a time via the more painstaking mine hunting (having a diver/ROV destroy the mines).
- Mine warfare vessels are expected to get banged up by the blasts as part of the job. For sustained scenarios, having a 1 /3 Rule mission and a port where minesweepers/hunters can be repaired is strongly recommended. A mission *without* the 1/3 rule can be used for an emergency surge if need be, but this is a game of endurance.

- Aircraft tend to be worse at detecting mines but better at destroying them, and the reverse is true for ships (thus they complement each other). While less vulnerable, low flying aircraft can still be damaged/destroyed by detonating mines.
- Unmanned vehicles are ideal for mine hunting due to less crew risk, but they have some limitations, such as having no one on deck to shoot floating mines.
- For mine clearing missions, a narrow safe corridor is better than a too-wide area that leaves too much room for mines to slip through.

9.2.5 Land Combat

Land combat in Command is currently very basic and works similarly to naval combat in terms of accuracy with gunfire or missiles. Land units are intended more as targets for air and sea units than as formations capable of credibly fighting each other on their own.

This should not discourage scenario writers, for land units can still be very important. They can still fire and move, and the role of target is an important one. The offensive capability of land units is not incompatible with a role as a target—after all, you might be tasked to destroy a group of “target” AFVs before they destroy your airbase or other high-value unit. (After all, as the saying goes, “the most effective air superiority weapon is a tank sitting at the end of the enemy's runway”.)

The terrain a land unit occupies (visible on the databox) has a massive effect on both its speed and its resistance. Swamps reduce movement dramatically. Forests slow movement down and provide a level of shielding from bomb/shell blast effects. Built-up areas add a high level of shielding but also make land units able to go faster due to their paved roads.

Land units have the capability of replenishing like UNREP-capable ships. For example:

A missile battery unit is near an ammo truck unit that has the appropriate missiles in its magazines. Once the battery has launched and is out of missiles, the player right-clicks on the battery unit and selects “replenish (if possible)” from the dialog. Whether the choice is to automatically or manually select the unit, the process will start. The missile battery will move alongside the ammo truck, stop, and restock. Once the process is complete, it may fire again.

When aircraft engage a land or naval target with unguided weapons, the calculation for hitting depends on the target's speed, the properties of the weapon, the bombsight quality of the aircraft, the altitude of the aircraft, and the proficiency of the aircraft's crew. A WWII-vintage fighter crewed by novice's nuisance-bombing at high altitude is going to be far less accurate than an ace plane with a high-end ballistic computer flying low.

9.2.6 Electronic Warfare

There are two types of electronic countermeasures in COMMAND: Defensive (DECM) and offensive (OECM).

DECM functions very simply. When a weapon with an appropriate seeker type closes on a unit with the DECM equipped, it functions as part of the weapon endgame calc. Whether the DECM can spoof the oncoming weapon depends on both random chance and the generations of both the DECM device and the seeker attacking weapon. An old DECM jammer will not have a good chance of stopping a weapon with a modern seeker and vice versa.

OECM projects out “noise” jamming. This has its strengths compared to the focused DECM. It can affect multiple units on the map, confusing search radars. However, it also has its weaknesses. First, ESM can often detect jamming as an emission unto itself, giving the position of the jammer. Second, it is very dependent on geometry and positioning, in addition to the technology level of both the jammer and target(s).

So if a jamming aircraft is supporting a flight of strikers, it’s highly advisable to position it directly behind them, on a course/mission that makes it stay as close to that spot as possible. It should also be as close as possible to the target while still comparably “safe”.

OECM does not currently affect endgame calcs directly the way DECM does, but (again, depending on the sophistication/strength of the jammer, the sophistication/strength of the opposing radar system, and the position of both), it can make SARH missile attacks fail due to disrupting the radars or make the target location so imprecise that the opponent never launches to begin with.

9.2.7 Damage and Repairs

A unit that is damaged but not immediately destroyed will suffer in terms of performance and may catch on fire, or (for ships) flood.

While one of them is active, the unit’s damage status will continue to increase, and if either of them reaches the maximum level, the unit will be automatically destroyed (it either burns out of control or becomes filled with water and capsizes), regardless of its damage status.

The crew will attempt to control the damage and repair it to the best of their ability, but their ability to do this in practice depends massively on circumstances, such as:

- Crew proficiency. As historical evidence shows, this can have a massive effect on whether or not a ship can survive a hit.

- How fast the unit is moving. Something strained by running at full or flank throttle will be more vulnerable.

9.2.8 My @#%^@#% weapon won’t fire!!!

COMMAND makes few (if any) compromises when it comes to the various factors affecting weapons and their effect on targets, as well as their employment. As a result, you may find yourself in a situation where it is not immediately obvious why the unit you just ordered to engage the enemy is not firing yet.

A very effective method of getting good feedback on weaponry limitations is to perform a manual weapon assignment. Your virtual crew runs the available weapons through an exhaustive checklist of conditions that (depending on the weapon and target) have to be met before the weapon at hand can be fired. Some weapons (such as wired guided ones) require that the launcher itself is not occupied by a new weapon as long as the current fired weapon is active. This means that reloading (and firing another one) is not possible until the situation is rectified.

Here is a summary of the no-fire explanations, what they mean and what you can do as a player to overcome them:

- **Weapon mount is not operational:** The weapon mount (gun, missile launcher, torpedo tube etc.) holding the weapon in question is out of action (probably damaged or destroyed) and cannot fire.
 - Solution: Be patient until the mount is repaired (if not destroyed outright). If possible, have the weapon unloaded from this mount so that it can be transferred to another suitable one, if such exists.
- **Unit is not authorized to use nuclear weapons:** The unit is attempting to fire a nuclear weapon but has no authorization to do so.
 - Solution: If you can (you are usually not allowed to), set the unit's "Use nukes" doctrine setting to "Yes".
- **Target speed [target speed in knots] is much higher than the weapon's maximum target speed ([weapon maximum target speed in knots]):** This applies mostly to SAMs and ABMs. Most such weapons have a limit to how fast a target they can intercept, either because of kinematic limitations or because their seekers and guidance systems can cope with only so much weapon-to-target closing speed. For example, most versions of the Patriot SAM can intercept tactical ballistic missiles, and the latest PAC-3 versions can tackle medium-range ballistic missiles, but ICBMs are a no-no.
 - Solution: None really, unless the target somehow reduces speed. Use another weapon if available.
- **Target altitude ([target altitude in meters]) is higher than the weapon's ceiling ([weapon maximum target altitude in meters]):** Quite simply, the target is flying too fast for the weapon to reach up and intercept it.
 - Solution: None, unless the target later descends. Use another weapon if available (there's a reason why, once more advanced low-range air defenses proliferated, attack aircraft began flying at high altitude.)
- **Target altitude ([target altitude in meters]) is lower than the weapon's minimum engagement altitude ([weapon minimum target altitude in meters]):** Most SAMs have a minimum engagement altitude below which their guidance systems suffer disproportionately from ground clutter and other factors. If the target is below that altitude then the weapon cannot engage.
 - Solution: Wait for the target to rise higher or use another weapon if available (there's a reason why flying low during the period of early SAMs with low ceilings was considered important).
- **Weapon is not BOL-capable:** The unit is attempting a BOL (Bearing-only Launch) shot but the weapon does not support this.
 - Solution: Execute a direct (non-BOL) shot or use another weapon that does have the ability.
- **Weapon needs a precise target location:** Related to BOL ability or lack thereof, this weapon needs a precise target location instead of an ambiguous estimate in order to have a reasonable chance of damaging the target. Guns usually fall into this category, as do other weapons like ASW rocket launchers.
 - Solution: Obtain a precise target location and use the weapon or use another weapon that is more tolerant of target ambiguity.
- **Weapon is not loaded on mount:** The weapon is present on the unit but not loaded on any of its mounts (e.g. stored in one of the magazines).
 - Solution: Wait for weapon to be loaded on mount; if necessary assign reload priority to it.
- **Weapon is not suitable for this target:** The weapon is generally not suitable for this target type (e.g. torpedo against aircraft)

- Solution: None; use another weapon if available.
- **Altitude too high (Valid: [weapon minimum launch altitude] to [weapon maximum launch altitude]):** Frequently happening with aircraft but can also apply to submarines. The unit is high above the weapon's launch envelope.
 - Solution: Dive until you are within launch parameters.
 - **Altitude too low (Valid: [weapon minimum launch altitude] to [weapon maximum launch altitude]):** Flip-side of above; the unit is too low/deep to use the weapon.
 - Solution: Rise up until you are within launch parameters. (Careful! You usually go deep/low for a reason - to stay out of enemy sensors/weapons. Coming up may mean attracting the wrong kind of attention.)
 - **Target is outside weapon boresight limits:** This is usually a problem with forward-firing weapons on aircraft (guns, rockets etc.) but may also manifest with fixed-azimuth (non-turreted guns).
 - Solution: Maneuver the unit so that the target is within boresight limits (typically dead ahead of the unit's front).
 - **Target aspect ([target aspect in degrees]) is out of envelope for a stern-chase weapon:** Most often a problem with early AAMs but also certain old torpedoes: The weapon is not very maneuver-friendly and needs to be fired almost directly behind the target.
 - Solution: Instead of the weapon, the unit will have to do the maneuvering; position it as much directly behind the target as possible.
 - **Target aspect ([target aspect in degrees]) is out of envelope for a rear-aspect weapon:** Similar to the above, but with less tight constraints.
 - Solution: As above; get behind the target (or off to the side) and launch the weapon if possible.
 - **Weapon cannot engage this target for another [time value in sec] (OODA loop limitation):** Each unit has a certain delay between first detection of a contact (or being given its data from elsewhere) and being able to target and engage it (the so called "OODA loop" – yes, the one from John "40-second" Boyd). With surprise threats this delay can often be fatal.
 - Solution: None really, just hope that the countdown runs out before the target becomes an imminent threat. If it is already an imminent threat (e.g. incoming anti-ship missile) and it looks like it's beating the clock, hope that the unit has automated point-defense systems that can ignore the OODA countdown and engage it. If not... brace for impact.
 - (NOTE: As a *scenario designer*, this problem can be mitigated by increasing the proficiency of the unit/side. Higher proficiency units have a shorter OODA clock)
 - **ASW torpedo must be dropped within 0.5nm of contact/aimpoint:** ASW torpedoes must be dropped close to the estimated target submarine position in order to have a fair chance of catching up to it before running out of fuel.
 - Solution: Get closer to the sub's estimated position and drop.
 - **Target is out of weapon's range:** Quite simply, the target is out of this weapon's maximum range.
 - Solution: Get closer (which may be easier said than done), or use a longer-ranged weapon.

- **Target is within weapon's minimum range:** Flip-side of the above; the weapon has a non-trivial minimum firing range and the target is within it.
 - Solution: Open the distance or use another weapon without the minimum range limitation.
- **Horizontal range to target ([range to target in nm]) is greater than the weapon's downrange at this altitude ([weapon downrange in nm]):** This most often happens with aircraft-launched ballistic weapons (guns, rockets, unguided bombs etc.). The higher the altitude of the carrier aircraft, the greater the distance at which it can literally toss the weapon towards the target. At higher altitude this distance shrinks dramatically.
 - Solution: Get closer or rise to higher altitude to increase toss range. Both these options carry counter-detection/engagement risk so choose carefully (or use another weapon with better standoff).
- **Target is within 5 nm and outside the weapon mount's engagement arc:** This typically applies to ship turrets/launchers. Command currently ignores mount arc limits for engagements at distances over 5nm; the implicit assumption is that at such distances captains micro-manuever their ships to unmask their weapons as necessary, and often even perform “over the shoulder” shots with guided weapons.

At distances below 5 nautical miles however, there is no time or space for maneuvers and shot-tricks; a ship engages a threat only with the weapons it can bring to bear down the threat's azimuth. This can make life difficult for large ships trying to engage smaller, more agile opponents or pop-up threats like sea-skimming anti-ship missiles.

- Solution: Try to turn around so the weapon you want to use is unmasked. If the target is a high-speed incoming threat (torpedo, missile e.g.) you have very little available time so this may possibly not work.
- **Cannot fire weapon through ice:** You are attempting to engage a target through the ice pack (e.g. shoot at a submarine with an air-dropped weapon).
 - Solution: None. If possible, wait until the target moves from under the ice before trying to engage again.
- **Cannot fire missile while under ice:** You are attempting to fire a missile from a submerged submarine under the ice.
 - Solution: None. Move the submarine out of the ice if possible.
- **Cannot use torpedo on ice:** You are attempting to drop a torpedo from an aircraft/helicopter to an under-ice submarine.
 - Solution: None.
- **Gun has no local control and no available directors:** The gun is designed to work with a fire-control sensor on the same unit which is currently out of action (for any number of reasons: Damaged/destroyed, jammed, cannot detect target etc.), and it (the gun) has no local/manual control as backup. It is essentially “blind”. NOTE: This limitation appears only if you have enabled the “Detailed gun fire control” realism setting.
 - Solution: If the FC director suitable for the gun is functioning but unable to detect the target, do anything necessary to allow it to do so (counter jamming, change heading to unmask director etc.). Otherwise use another weapon if possible.
- **Weapon must detect target prior to firing:** The weapon must “lock on” to the target on its own prior to launch, and cannot be fired “blindly” with the expectation of post-launch lock. Most IR-guided AAMs fall in this category.

- **Solution:** Position the firing unit in optimum position for the weapon to detect the target (e.g. in a dogfight, try to get on the target's tail as much as possible). The weapon will do the rest.
- **No weapons director available to illuminate the target:** The guided weapon (usually missile) needs a compatible FC director to provide illumination to the target in order to guide the weapon, and no such sensor can be found (in an operative state). Similar to gun-fire control limitations.
 - **Solution:** If the FC director suitable for the gun is functioning but unable to detect the target, do anything necessary to allow it to do so (counter jamming, change position to obtain line-of-sight etc.). Otherwise use another weapon.
- **All illumination channels suitable for this weapon are in use:** Related to the above; we found a suitable FC illuminator for the weapon but all available illumination/guidance channels are in use (ie, the system is already locked on to its maximum number of targets and can't engage any more).
 - **Solution:** This indicates that one or more engagements are currently in progress. Wait for the engagement(s) to conclude so that the illumination/guidance channels can be freed up.
- **All directors are unable to illuminate this target (insufficient reflection, no LOS etc.):** The guided weapon (usually missile) needs a compatible FC director to provide illumination to the target in order to guide the weapon, and no such sensor can be found (in an operative state). Similar to gun-fire control limitations. This is frequently observed with guidance radars against low-observability targets (in fact, the F-117 was created to be resistant to guidance radars first and search radars second).
 - **Solution:** If the FC director suitable for the gun is functioning but unable to detect the target, do anything necessary to allow it to do so (counter jamming, change position to obtain line-of-sight etc.). Otherwise use another weapon or see if the target moves closer.
- **No datalink channel available to guide this weapon:** The weapon has a mandatory datalink (such as the guidance wire on a TOW anti-tank missile) and no channel is currently available.
 - **Solution:** This indicates that one or more engagements are currently in progress. Wait for the engagement(s) to conclude so that the datalink channels can be freed up.
- **Target is out of weapon's DLZ:** The target is out of the weapon's dynamic launch zone (DLZ; see "DLZ and why it's important" in the Appendix at the end of this manual).
 - **Solution:** Get closer to the target or wait for it to change its heading/speed/alt so that it falls within the weapon's DLZ.
- **The target's downrange ambiguity ([weapon downrange ambiguity in nm]) is larger than [some percentage] the weapon's acceptable limit ([weapon acceptable downrange ambiguity in nm]):** This is directly related to the "Shooting ambiguous targets" doctrine setting. If the target's uncertainty area is such that its range ambiguity towards the firing unit is too large, the weapon will not fire. For example if the target's distance has a +/- 10nm margin of error, and the weapon has an acceptable limit of 2nm (seeker limit, warhead damage area etc.) then the unit will refuse to fire.
 - **Solution:** If you can (you may not be allowed to), set the firing unit's "Shooting ambiguous targets" doctrine setting to "Optimistic" or even "Ignore". This will make the unit all too happy to use the weapon against the target even with a large margin of error. Beware that this means a large probability that the weapon will miss altogether. Ignore is not recommended against targets that have visibly moved at all.

- **The target's downrange ambiguity ([weapon downrange ambiguity in nm]) is larger than [some percentage] the weapon's acceptable limit ([weapon acceptable downrange ambiguity in nm]):** As above, this time with regards to target cross-range instead of distance ambiguity.
 - Solution: If you can (you may not be allowed to), set the firing unit's "Shooting ambiguous targets" doctrine setting to "Optimistic" or even "Ignore". This will make the unit all too happy to use the weapon against the target even with a large margin of error. WARNING: Large angular margins of error mean that you have to be really lucky to get a successful hit.

9.2.9 DLZ And Why It Matters

When firing a guided weapon against a target, we want to have a reasonable probability (if not certainty) that the weapon will reach the target. Against static or slow-moving targets this is a straightforward affair; against high speed and/or agile targets, however, things can get more complicated.

One of the simplest concepts of estimating a weapon's reach against a moving target is the so-called "No-Escape Zone" or NEZ. This assumes that at the moment of weapon fire, the target turns tail and runs at its current known speed directly away from the launch point. The weapon then has to run the target down in a straight line.

The simplicity of NEZ made it attractive for early missile fire-control computers and indeed it remains a widely-used yardstick when comparing the kinematic performance of weapons such as AAMs/SAMs, torpedoes etc. However, in real combat units rarely detect incoming weapons immediately at launch and not often turn tail and run away – at least not if they have a job to do. The NEZ is also meaningless against targets that naturally don't run away, such as cruise missiles and UAVs (and non-alerted targets of any kind, e.g. very frequently ships under torpedo attack).

Relying exclusively on NEZ is can also represent a "hidden" tactical liability. Consider an F-15 fighter preparing for a Sparrow AAM shot against a high-speed incoming MiG-25. Following NEZ's logic, the MiG's high speed means that the missile must be launched at very close range in order to successfully perform the hypothetical run-down that the NEZ is built upon. However, we instinctively understand that this logic is flawed: The F-15 pilot should instead launch the Sparrow at an even longer-than-nominal range, taking advantage of the MiG's high incoming speed and using it against it.

So what we need is an engagement zone concept that takes into account additional parameters known about the target such as its current heading and altitude. This concept is the Dynamic Launch Zone or DLZ.

Fundamentally, the conceptual difference between NEZ and DLZ is clear. NEZ answers the question: What will happen if I launch right now and the target turns right away and runs? DLZ answers the question: What will happen if I launch right now and the target keeps moving as-is?

The DLZ is harder to calculate for two reasons.

The first one is fairly obvious: Unless the target is coming head-on or running straight away, the engagement geometry is significantly more complex. The second reason is that in order to answer the question faithfully, the algorithm must consider the kinematic & guidance idiosyncrasies of the weapon in question. Some weapons head straight for their target at a near-constant speed; others deliberately opt for a parabolic loft; yet others employ subtle guidance tricks tailored to the nature of both the weapon's role and the characteristics of

the nominal target the weapon was designed for. All these nooks and crannies must be taken into account or else the DLZ calculation may well be off.

Command employs several algorithms for faithful DLZ calculations, some tailored for aerodynamic targets (with special options for non-maneuvering targets such as cruise missiles) and others optimized for orbital & ballistic targets such as ballistic missiles, re-entry vehicles and satellites. You will notice that AI crews are usually quite careful not to fire weapons at the extreme edge of their range envelope, preferring to sacrifice a slice of standoff for the sake of ensuring a kill. There still will be outliers though, taking their chances with long-range “sniper” shots that can still do damage.

It should be emphasized that in the case of maneuverable targets, being inside the DLZ does not guarantee that the weapon will reach out to the target – after all, an alerted enemy may (probably will) not cooperate. But even if a shot is dodged, it still puts the target on the defensive. The top Serbian MiG-29 pilot in the Kosovo conflict was finally shot down after successively avoiding 3 AMRAAM shots in a single engagement – each time surrendering more of the initiative to his attackers, never having a chance to shoot back.

9.2.10 A Note On Losses

No unit is indestructible, and operational planning frequently matters far more than the exact quality of the unit. It is entirely possible for a MiG-21 to destroy an F-15 if it gets a clear shot.

Much of the “turning/burning” parts of air combat are abstracted by hit rates and crew proficiency. The part of the human element the player can control is getting the assets to the right place at the right time. An example of the dichotomy between tactical weakness and operational strength is in the 1972 air campaign that beat back the Easter Offensive in the Vietnam War.

In that, the kill/loss ratio of USAF aircraft reached new lows, with them having a negative rating even by their own admission during various points (meaning the true total was almost certainly even worse). Yet this mostly consisted of North Vietnamese MiGs nibbling away at the outer escorts-despite their weaknesses, they did their job of keeping the strikers safe. In Command, you can do the same.

Even in scenarios intended to re-enact historical battles or wars, the results are frequently much different from the actual outcome. This is to be expected. While a bug report (see 1.2, Support) may be in order if the historical result is *never* obtained even once, variation is normal and good.

Command’s engine is intended to be able to simulate everything from 1940s broadside duels to futuristic satellite-cued ballistic missile strikes with a reasonable degree of success, not be incredibly accurate for any one conflict. To return to the Vietnam analogy, the performance of similar, and frequently the exact same aircraft and weapons varied widely by conflict. There was Vietnam, the Indo-Pakistan, and Arab-Israeli Wars in the same time period.

Attempting to model any one of these conflicts with one hundred percent razor-sharp accuracy would make any of the others less effective. One would be faced with either an ahistorically strong Egyptian/Syrian Air Force or an ahistorically weak North Vietnamese one if a strict model intended for one was used for the other.

In addition, real events can only happen once, while the simulator can be run many times. The real result may have been a one-in-a-hundred “outlier”, while the most common one may have been quite different.

9.3 Construction and Destruction

9.3.1 Building and Destroying Air Bases.

Air Bases are important units in COMMAND. They can be added to scenarios either as a single unit or air base group. At the basic level they host aircraft but they also support the simulations advanced aircraft logistics and sortie models and act as advanced targets. This section explains how to build these units and how to destroy them.

Air bases are currently implemented in the game as either one very generic single-serving unit (Single-Unit Airfield) or as an airbase group that could include hundreds of individual facilities. As a designer you can choose the type you would like to use but unit-count does impact performance so it's best to use a single-unit airfield unless the air base has the potential to be a target. There is no difference in function between the two. COMMAND'S databases provide a ready selection of such bases in various sizes (going all the way up to multiple 4000-m runways) so you can select what most closely fits your needs.

COMMAND'S air bases do model aircraft traffic, which helps support the simulation accurately model aircraft-sortie rates. When given a launch order, aircraft will start in their holding facilities such as tarmacs, hangars or revetments, move to a connecting/transit facility such as a Runway Access Point or Taxiway (or elevator in a ship) and then move to a Runway where they are launched. When they land, they step back through the flow to their holding facility, receive replacement fuel and ammunition and then start the process again. This helps generate realistic sortie rates which we feel are critical in modelling modern air warfare. It also makes airbases critically dependent on a few key nodes being operational – or, put from another perspective, if an enemy damages or destroys certain facilities he can shut down an airbase with much less effort, just like in real life.

Air bases currently support the aircraft logistics model in the game. All air bases include ammunition magazines which players can populate with the weapons, pods, tanks and equipment that aircraft need to successfully host their various loads. Fuel is not currently tracked but is provided automatically when an aircraft lands. As such, scenarios that exceed the flight time or an aircraft must include an air base or a ship capable of hosting aircraft or the aircraft will ditch (and be destroyed) when its fuel has been expended.

For more information on how to populate ammo magazines please review section 5.3.1 of the manual. It is highly suggested that you use the “Show only weapons compatible with aircraft hosted by parent” filter in the Add Weapons dialog to assist with finding the correct weapon-records to add.

Constructing an Air Base

Constructing an air base group is accomplished in a similar manner to building other groups in the game. You should add the individual facilities and then group them by selecting them all and pressing **the g key**.

It is highly suggested that you use our “Custom Overlay” feature to get a properly sized and sited image of the facility you are adding on your game map. This allows you to simply add the needed units where you see them in the image – easily and with extreme accuracy.

Once you have constructed the base, you can then use the import/export unit feature to save it for future use. We have already constructed thousands of airfields, airbases and airports which you can import into the game so it is best to check first before spending too much time building your own.

The databases have tons of air base facilities but the minimum units necessary for a functional air base group are as follows:

- 1x Runway long enough to support the aircraft you will base there. You can get this by comparing the runway length in the name and database viewer entry with the aircraft Take-off/Landing distance field in the database viewer.
- 1x Access Point or Taxiway large enough to support the aircraft that will be using them.
- 1x Unit Capable of holding the aircraft (Tarmacs, Open Parking Spaces, Hangars, Shelters, etc.) you'd like to host there. Units have the aircraft size and capacity in their names which you can compare with what is listed in the aircraft size field in the database viewer.
- 1x Unit with an ammunition magazine (Ammo Shelters, Ammo Pads, Ammo Bunker, etc.)
- 1x Fuel holding facility. We aren't requiring this as we are not tracking fuel logistics but may do so in the future.

Adding Single-Unit Airfield airfields is straight forward in that you just add them as you would any other unit in the game. They include all the necessary pieces required to function and just require you to rename the unit to whatever you would like unit to be. These can be saved to import file and exported back in as well.

Be aware that single-unit airfields are not intended to be targets and should only be included in a scenario where they will not be attacked. Thus a forward airstrip in the heat of the battle should be modelled as a multi-unit airbase, while a base hosting long-range bombers far away from any of the opponent's weapons should be modelled as a single-unit one.

Destroying or Inhibiting Air Base Operations

Tying everything we have learned about the functions of Command air bases and how to build them, we can now see ways to destroy or mission-kill them.

Keep in mind that unless the air base is small enough and/or the attacker strong to simply smother it, determining where the individual aircraft are is essential. For aircraft parked in the open, this is as simple as moving a unit with optical sensors close enough to detect and/or identify them. Under the "contact reports" window, the detected aircraft will appear under the "Spotted Hosted Units" tab, they will appear on the message log when detected/identified, and on the map, a black triangle in a yellow box will mark any facility with hosted units inside.

For enclosed shelters, things get trickier. The aircraft have to be spotted entering or leaving them over the course of air operations to get an identification. This necessitates either persistent surveillance (like a ground recon team) or good timing.

Methods to Mission-kill or otherwise incapacitate a modern air base:

- **Destroy aircraft holding facilities:** Probably the least efficient but you can eliminate all aircraft at the facility by destroying the buildings/sites that contain them. Most nations harden these facilities or disperse aircraft to make this task difficult. To evaluate a weapon's effectiveness against a target please review the weapon's legal target values and damage point value (DPs) in the aircraft stores section of the database viewer and compare it to the target's armor and damage point values.

- **Destroy Magazines:** Destroying ammo storage facilities prevents aircraft from receiving necessary ammunition. A combat aircraft without its weapons is only marginally more useful than if it has been destroyed. Similar to aircraft holding facilities, most nations harden the facilities or disperse ammunition to minimize losses. Note that this will not work if unlimited ammunition is enabled in scenario settings, so if destroying airbase magazines is important, the scenario designer should turn that option off.
- **Disable Access Points, Taxiways and Elevators:** Disabling (these facilities cannot be permanently destroyed and will eventually be repaired given enough time) these “transit” points prevent aircraft from reaching runways or other aircraft launch facilities. Usually there are only a few of those in a base or ship so taking them out is a great way to jam the base shut.
- **Disable runways:** Disabling these eliminates the ability of an aircraft to launch. This is often the most efficient method. Nations tend to add secondary runways or use auxiliary runways to make this task slightly more difficult. COMMAND supports numerous dedicated anti-runway weapons. It should be noted that historically, runway attacks have only been useful as “temporary” knockouts to airbases. The runway can be repaired quickly just by repaving it, especially if it’s been hit by non-specialized munitions. As such, many planners have aimed for targets other than runways to cause more lasting damage.

9.3.2 Building and Destroying Naval Bases

Naval bases are like air bases but require only two facilities for a minimum functional group: at least one pier and a munitions storage facility. Expansive facilities are encouraged for the sake of realism and targeting-big ships require big ports.

Docked naval units at ports can be detected and identified like aircraft at air bases. The same criterion for observation is there, as is the same black and yellow symbol indicating a present unit.

9.3.3 Building (And Destroying) Air Defenses

Air defense facilities in COMMAND are some of the most important land units. Anti-aircraft artillery, and, since the Powers U-2 shootdown, surface-to-air missiles have shaped the landscape of air operations.

- **SENSORS:** The air defense network needs to be able to see its incoming targets before it can hit them. This includes search radars as the most obvious component, but a truly effective IADS will also have ESM sensors and even visual observers. The search radars are frequently kept a considerable distance apart from the launch sites.
- **WEAPONS:** Anti-aircraft weapons come in three major categories: Low altitude, medium altitude, and high altitude. They combine to become more than the sum of their parts when properly cited, with the most famous example being the North Vietnamese SAMs that did not score too many kills on their own- because they made the Americans fly lower, right into the teeth of the AAA that claimed the most victories.
 - Low Altitude/Short Range: This includes almost all anti-aircraft guns and MANPADS like the Stinger and SA-7.
 - Medium Altitude/Range: This includes medium-sized missiles like the SA-6 and Aspid.
 - High Altitude/Range: This includes heavy missiles, whether they be early (SA-2 and Nike) or modern (Patriot and S-300 family)

Air defense facilities should cover as many altitudes and angles as possible. They should be mutually supporting, covering both the defended area and each other. Placing short range point defense systems around both longer-range launchers and search radars is recommended, particularly if the opponent has standoff weapons like Tomahawks. Radars should not emit unless absolutely necessary.

For the destruction of air defenses, there are two options. The first is to simply treat them like any other target, engaging them with conventional strikes and patrols. The second is dedicated SEAD patrols, which focus on radar emissions. There is no one-size fits all solution to dealing with air defenses.

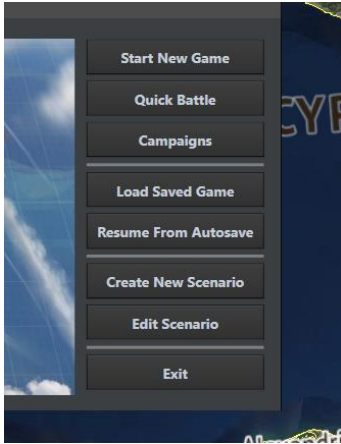
9.4 Unmanned Aerial Vehicles

UAVs have become a dominant and omnipresent force on the battlefield. COMMAND models some of the special functionality of the UAVs, especially how they can operate themselves when they lose contact with their signal provider. Some UAVs, nothing more than remotely controlled aircraft, will on loss of signal become “ghost planes,” flying off on their last heading until eventually crashing. Some can self-recover, i.e. search for a new signal and return to base if reconnection proves impossible. Newer UAVs are proving increasingly more autonomous, able to complete their last orders, self-analyze their ability to continue, work together, or even “think” for themselves. Once this is implemented, jamming drones will have platform-specific consequences beyond simply losing communications: some drones will crash immediately, while others may go off and hunt down John Connor.

10.0 Example Scenario Play

10.1 How to Start a Scenario

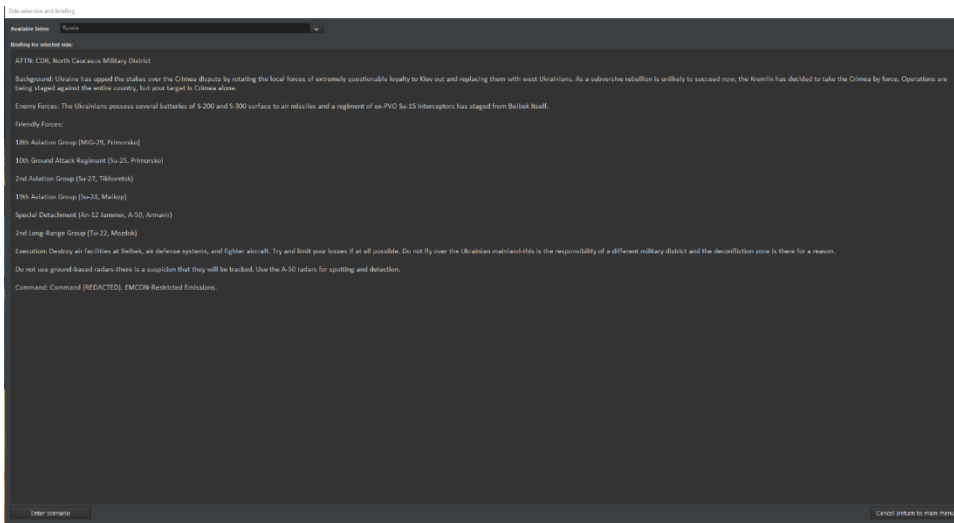
Proceed to the main menu and select “Start New Game”



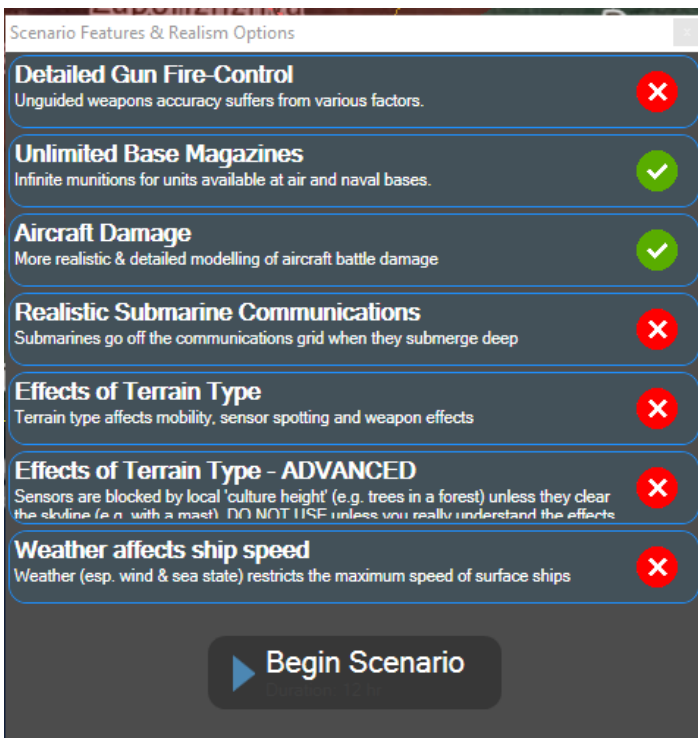
Then select the name of the scenario



The ratings for complexity and difficulty are set by the designer of the scenario and are variable depending on many factors. Generally, scenarios that are more difficult tend to have serious tactical challenges or a very active threat environment. Scenarios that are more complex tend to both be longer, have more domains, numbers of units, and rules of engagement limitations. After selecting a scenario and pressing Load selected, a menu will pop up providing you with a detailed briefing as well as selectable sides.



Following this, a menu will pop up notifying you of the various features and realism options set by the mission designer. These can be tweaked in the mission editor, but are otherwise provided for user information only.

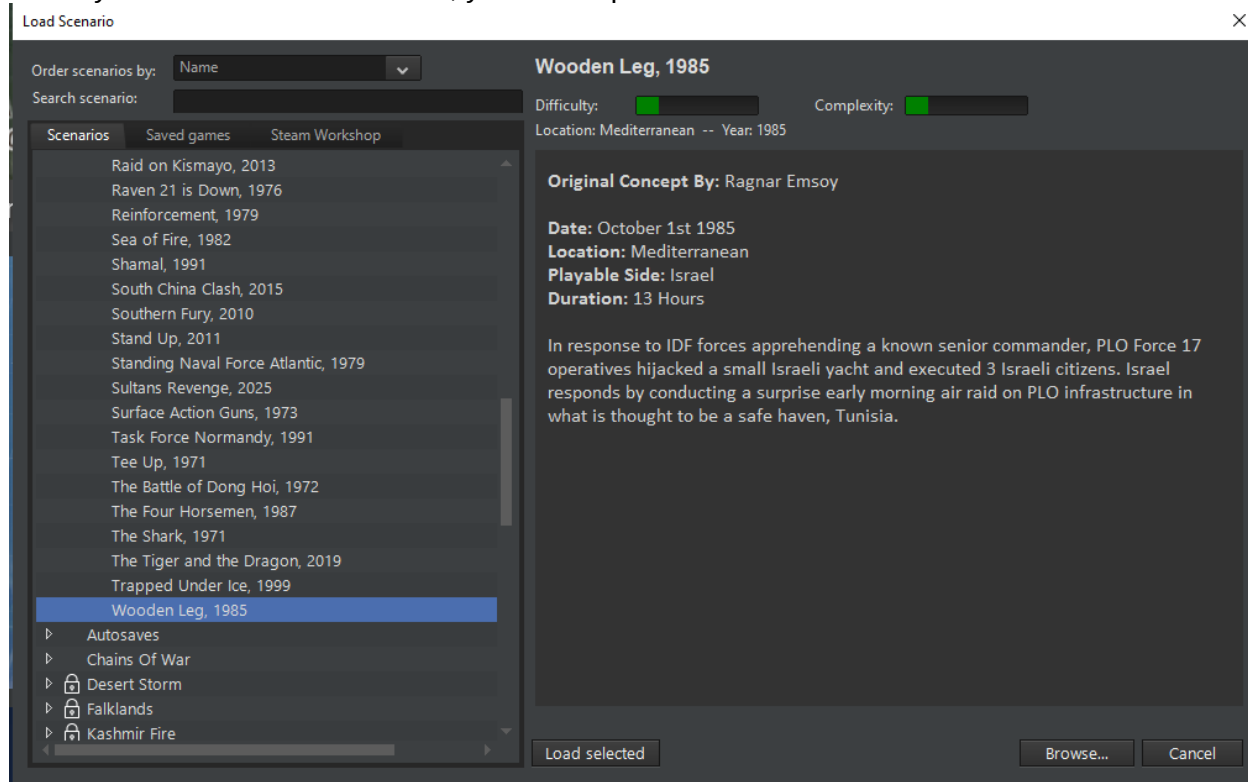


10.2 Wooden Leg, 1985

10.2.1 Collecting Intelligence and Devising our Strategy

This scenario provides a great starting point for new players to COMMAND as the complexity is relatively low but in order to achieve the greatest result, many different techniques and strategies must be employed.

When you first select the scenario, you will be provided with a few different details:



We are given a general overview of where, what and how we should execute the scenario. After pressing load selected, we are given a much more detailed briefing.

Available Sides:
Israel

Briefing for selected side:

Situation

PLO Force 17 operatives have murdered Israeli citizens while their leadership continues to direct their activities from the refuge of Tunisia. We must prove that there is no refuge for terrorists.

Enemy Forces

Palestinian Liberation Organisation

PLO forces are likely at high alert but have no ground to air weapons capable of engaging our forces other than small arms fire.

Tunisia

Tunisian forces do have several F-5 fighters based at Sidi Ahmed Air Base but are not likely to detect or deter an incoming raid. Your forces should be out of the area before any response can be mounted.

Air

- F-5A Freedom Fighter

Friendly Forces

Two KC-707 Saknayee tankers are already on station over the Mediterranean to provide refuelling for the outbound leg. An additional two KC-707s are on ground standby at Ben Gurion.

Six F-15Ds from 106 Sqn 'Spearhead' are armed with GBU-15 glide bombs for the main raid.

Two F-15Ds are armed with Mk82 bombs as backup.

A helicopter-carrying vessel (340 Aliyah) is stationed near Malta to recover downed pilots.

Surface

- 340 Aliyah [Saar 4.5, Helo Pad]

Air

340 Aliyah

- Det A (1 x SA.366G Dolphin)

Tel Aviv-Ben Gurion

- 120 Sqn (4 x KC-707 Saknayee)

Tel Nof Air Base

- 106 Sqd (8 x F-15D Akef)

Mission

Conduct a precision strike on PLO infrastructure in Tunis.

Execution

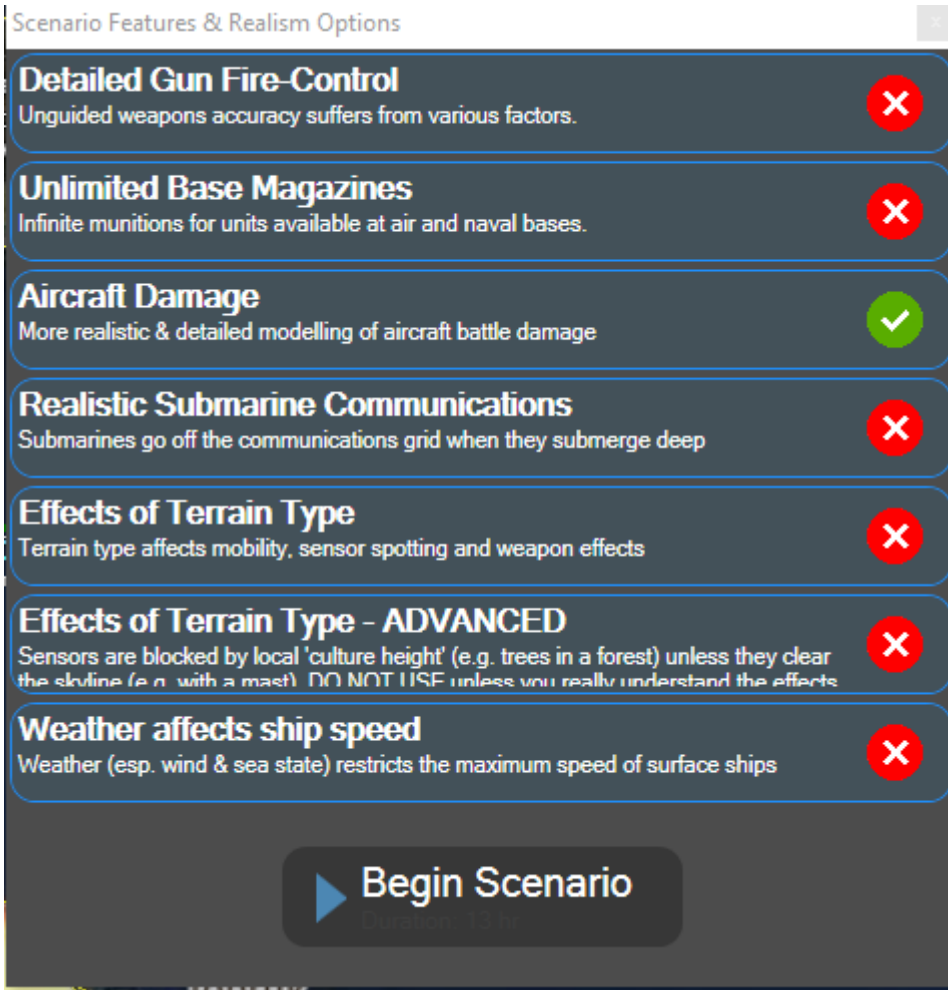
- Launch F-15D Akef aircraft from Tel Nof Airbase and vector towards the targets in Tunisia.
- Rendezvous enroute with KC-707 Saknayee aircraft to refuel
- Strike targets with GBU-15 CWW glide bombs
 - Utilise Mk82 equipped aircraft to neutralise any remaining targets after GBU-15s have been expended
- Return to base
 - Rendezvous with KC-707 Saknayee aircraft for refuelling on return leg if required

Enter scenario

The key things a player will be interested in will be the nature of the mission as well as the ways that it should be carried out. There are a few things that are relevant to us right away:

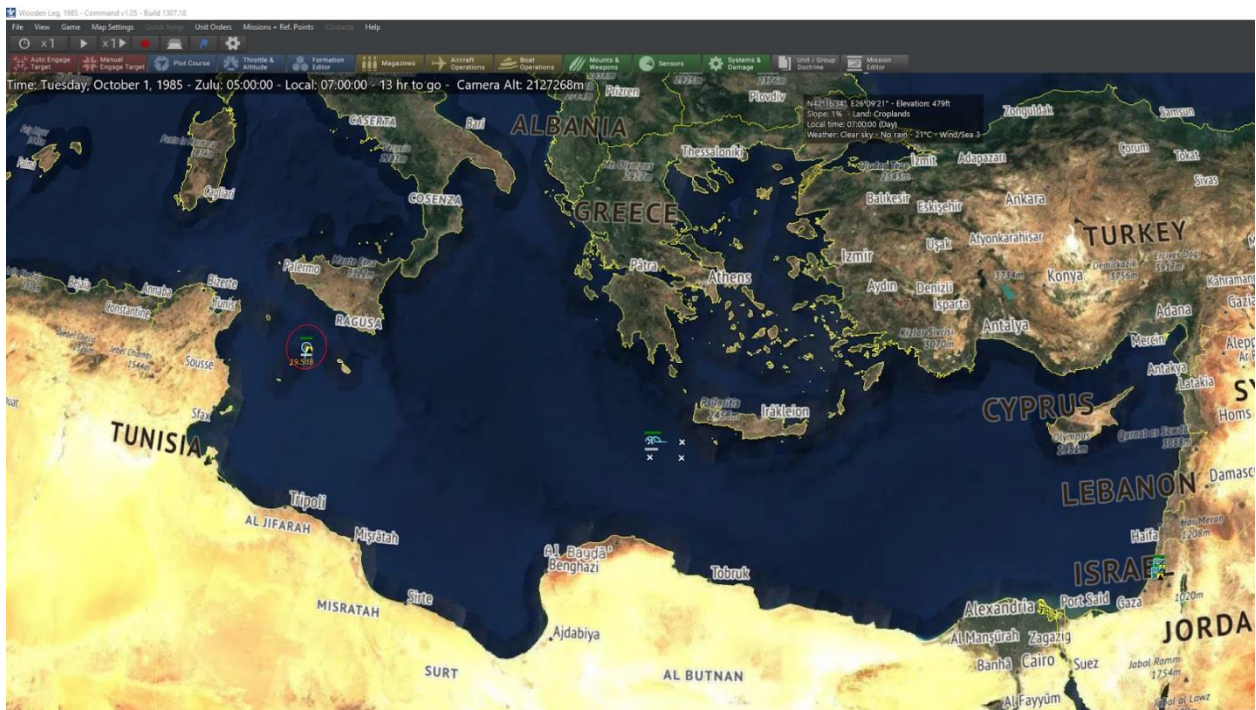
1. The objective is to destroy PLO buildings in Tunisia
2. The enemy resistance is expected to be light
3. We are to use the laser guided bombs to do maximum damage to the targets and *then* follow up with Mark 82s after the targets have received their initial damage.
4. The ranges of the mission will be long, requiring us to conduct refueling possibly in both directions.

On the next page we get an idea of the different scenario options that will affect us tactically.



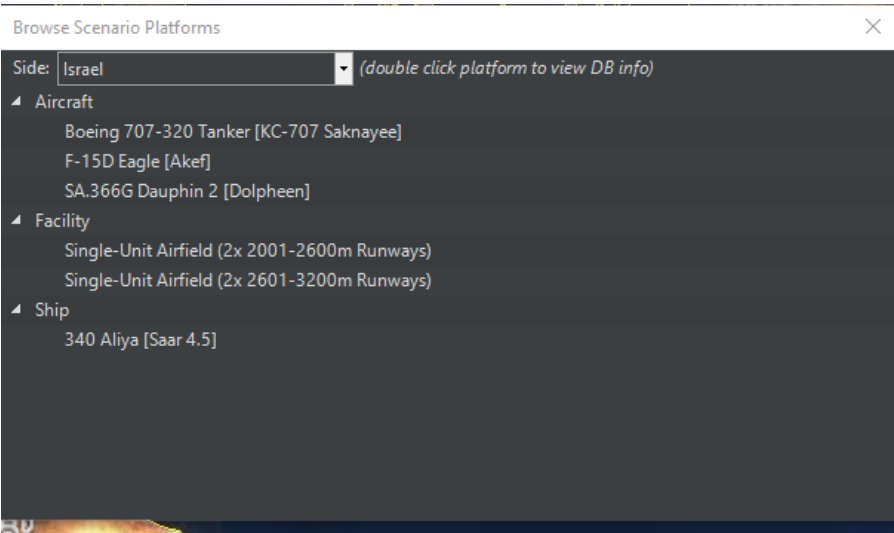
We can see right away that the only real concern we have is aircraft damage which means if our aircraft are hit lightly, they can still safely return to base. Many of the other options are not present in this scenario because they are either not required or are not relevant.

After we click Begin Scenario, we should pause and collect some intelligence.

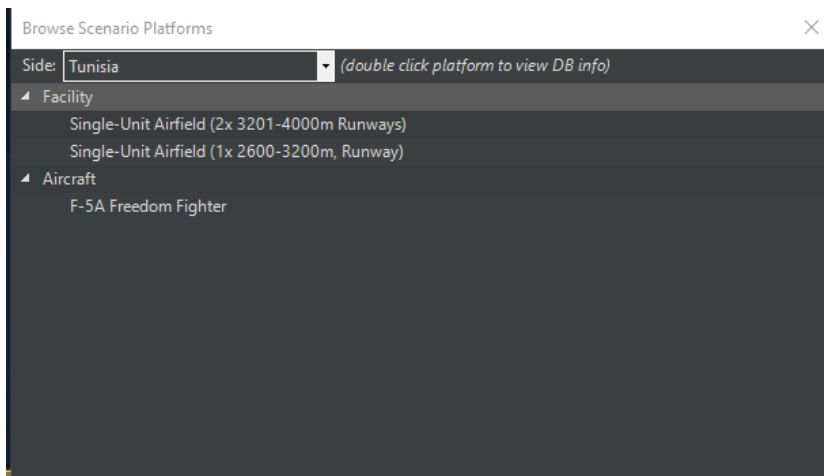


We can see that our scenario bounds range from Israel all the way to Tunisia and that there are no exclusion zones or no navigation zones anywhere on the map. This means we have free reign where we would like to operate our units throughout the scenario. Keep in mind that these requirements may be spelled out in more detail in a briefing or may be implemented in other ways.

We now would like to see what kinds of units are present in our scenario to see how this may affect our strategy. We will click Game -> Browse Scenario Platforms and take a look at what we have at our disposal and what we can expect throughout the rest of the scenario.



Looks like we have some F-15Ds, a few rescue helicopters and a Patrol ship.



The PLO have a handful of buildings and we can also see that Tunisia has a few F-5As that may give us some trouble. Let's double click on one of them and see what we can find out.

GENERAL DATA

Category:	Fixed Wing	Aircraft size:	Medium Aircraft (12.1-18m)
Type:	Multirole (Fighter/Attack) (WRA: Aircraft - Increased manoeuvrability fighter (F-4, F-5, MiG-21, MiG-23))	Agility:	3
Length:	14.4 m	Average Climb Rate:	14390 ft/min, 73.1 m/sec
Wingspan:	7.9 m	Instantaneous Climb Rate, S/L:	43180 ft/min, 219.3 m/sec
Height:	4 m	Take-off/Landing Distance:	901-1400m TOD/LAD
Crew:	1	Cockpit Visibility:	Forward: Excellent Sideways: Excellent Aft: Average
Cockpit:	Steam Gauges (Complex) (ex: MiG-21)	Armor:	Fuselage: None Cockpit: None Powerplant: None
Empty Weight:	3670 kg	Damage Points:	5
Max Weight:	9307 kg		
Max Payload:	3087 kg		
OODA cycle:	Detection: 15 seconds Observe, Orient, Decide, and Act (reaction time) Targeting: 30 seconds (Novice Proficiency Level) 23 seconds (Cadet) 18 seconds (Regular) 15 seconds (Veteran) 12 seconds (Ace) Evasion: 2 seconds		

SENSORS / EW

Model	Max Range	Notes	Abilities	Arcs
1x Hki Eyeball	15 / 50 nm		Air Search, Surface Search, Ground Search (Fixed), Ground Search (Mobile), Range Information, Altitude Info, Heading Info	

MOUNTS / STORES / WEAPONS

Mounts (Guns/Launchers/Ejectors/etc.)

Mount	Capacity	Launch Interval	Armor	Onboard Sensors	Weapons (per mount)	Notes	Arcs
1x 20mm M39 x 2 [280 rmds x 2]	7	5	None		7x 20mm M39 x 2 Burst [80 rmds] Gun Targets: Aircraft, Helicopters, Surface Ships, Land Structures (Soft), Mobile Units (Soft), Missiles & Guided Bombs. Max Range: 1 nm. Warhead: 20mm Burst [80 rmds].	Local control possible.	

Aircraft Stores

Store	Speed Release Envelope	Altitude Release Envelope	Description
150 USG Drop Tank			Drop Tank
50 USG Drop Tank			Drop Tank
AIM-9N Sidewinder	200 ft AGL - 65000 ft ASL 60.9 m AGL - 19812.0 m ASL		Guided Weapon. Targets: Aircraft, Helicopters, Missiles & Guided Bombs. Max Speed: 1475 kts. Max Range: 8 nm. Warhead: AIM-9B/C/D/E/F/J/N/P 4.5kg Blast-Frag.
HYDRA 70mm		0 ft ASL - 10000 ft AGL	Rocket. Targets: Surface Ships, Land Structures (Hard), Land Structures (Soft), Mobile Units (Hard), Mobile Units (Soft). Max Range: 2 nm. Warhead: M229 2.2kg HEDP.

It looks like they can carry up to 4 AIM-9N Sidewinders and have a 110 minute CAP time. Depending on how many of the aircraft they have on the ground, they may be able to put up a continuous combat air patrol over the target region if they have enough warning. Luckily there are no EW radars in the scenario so it will be to our advantage not to use our own radars in the event they do scramble.

The AIM-9N is an interesting weapon as well:

Length:	3.05 m	Climb Rate, S/L: 1211 ft/min, 369.0 m/sec	Range (anti-surf): -	Surface PoH / Reliability: -
Span:	0.58 m	Launch Altitude: 200 ft AGL - 65000 ft ASL, 60.9 m AGL -19812.0 m ASL	Range (anti-land): -	Land PoH / Reliability: -
Diameter:	0.13 m	Launch Speed: -	Range (anti-sub): -	Sub PoH: -
Weight:	77 kg	Target Altitude: 30 ft AGL - 65000 ft ASL, 9.1 m AGL - 19812.0 m ASL		CEP, Surface (nominal): -
Burnout Weight:	19 kg	Target Speed: 0 - 1350 kt		CEP, Land (nominal): -
Guidance Type: Rear-Aspect mid-course plus Passive Infrared (IR) terminal guidance				

SENSORS / EW

Model	Max Range	Notes	Abilities
1x IR Seeker	25 / 10 nm	Weapon Seeker, Single Spectral IR	Seeker, Single Spectral IR Technology, Air Search, Altitude Info

SIGNATURES

Signature Type	Front	Side
Visual Detection Range	0.80 nm	0.80 nm
Infrared Detection Range	4.50 nm	4.50 nm
Radar, A-D Band (30-2000 MHz)	-16.20 dBsm, 0.024 sq.m.	-13.00 dBsm, 0.050 sq.m.
Radar, E-M Band (2-100 GHz)	-16.20 dBsm, 0.024 sq.m.	-13.00 dBsm, 0.050 sq.m.

PROPERTIES

Anti-Air Rear-Aspect

WARHEADS

#803 - AIM-9B/C/D/E/F/J/N/P 4.5kg Blast-Frag (1.58 DP)
--

VALID TARGETS

Aircraft
Helicopters
Missiles & Guided Bombs

The key element is that it is a rear aspect only missile meaning it can be used in a chase but not in a head to head engagement. This will have a profound strategic impact in our WRA later on in the game.

There are a few other things we can determine from our scenario, and that is that we have aerial refueling aircraft in position and we already have patrol craft ready in the event we need to do pilot recoveries.

The last thing we should check is the weather, which we can do by looking at the side of our cursor at any time. We can see that the skies are clear, the temperature is 21C and the wind/sea state is 3. This is good since it means we'll have excellent visibility all the way to the target and not have to fly below a cloud layer to employ our weapons.

Now let's get an idea of what our strikers look like. Click on Tel Nof Air Base and press F6, or click the Aircraft Operations button at the top of the screen.

Air Ops - Tel Nof Air Base

Aircraft Status

Air Facilities

Aircraft (click for info)	Status	Mission	Loadout	Time to ready	Quick Turnaround
8x F-15D Eagle (Akefi) (Tel Nof Air Base)					
106 Sqd #19	Parked	-	GBU-15(V1)/B CWW EO (Mk84), AN/AXQ-14 Datalink Pod, Long...	Ready	-
106 Sqd #20	Parked	-	GBU-15(V1)/B CWW EO (Mk84), AN/AXQ-14 Datalink Pod, Long...	Ready	-
106 Sqd #21	Parked	-	GBU-15(V1)/B CWW EO (Mk84), AN/AXQ-14 Datalink Pod, Long...	Ready	-
106 Sqd #22	Parked	-	GBU-15(V1)/B CWW EO (Mk84), AN/AXQ-14 Datalink Pod, Long...	Ready	-
106 Sqd #23	Parked	-	GBU-15(V1)/B CWW EO (Mk84), AN/AXQ-14 Datalink Pod, Long...	Ready	-
106 Sqd #24	Parked	-	GBU-15(V1)/B CWW EO (Mk84), AN/AXQ-14 Datalink Pod, Long...	Ready	-
106 Sqd #25	Parked	-	Mk82 LDGP, Long Range, 1985 Tunis Raid Backup Configurati...	Ready	-
106 Sqd #26	Parked	-	Mk82 LDGP, Long Range, 1985 Tunis Raid Backup Configurati...	Ready	-

☐ Enable Quick

Collapse aircraft list

Hide non-available

Occupied 240 / 3790 (6%)

Loadout Details: GBU-15(V1)/B CWW EO (Mk84), AN/AXQ-14 Datalink Pod, Long Range, 1985 Tunis Raid Main Configuration, Loadout Role: Standoff Strike, Land/Naval

Stores (click for info)

	# Available, Magazines	# Available, Mags + A/C
2x AIM-7M Sparrow III (Optional)	0	16
2x AIM-9L Sidewinder	0	16
1x GBU-15(V1)/B CWW EO (Mk84)	0	6
1x AN/AXQ-14 Datalink Pod	0	6
1x 600 USG Drop Tank	-	-
2x 750 USG CFT	-	-

Capabilities: Day-only, Limited all-weather capable
Range and Profile: 900 nm Strike radius, Hi-Hi-Hi profile. Cruise at 36k ft.
Weapon release at max range. Weapon release at max range. Form-up 10
Pre-briefed weapon state Winchester Return to base when mission-specific weapons have been expended. Disengage immediately.
Attack Altitude: 36000 ft ASL
Loadouts available in magazines: 0
Loadouts available, incl. weapons mounted on all aircraft: 6
Loadouts available, same as above excl. optional weapons: 6

Launch individually

Launch as group(s)

Ready / Arm

Abort Launch

Doctrine

Assign to mission

Cargo Ops

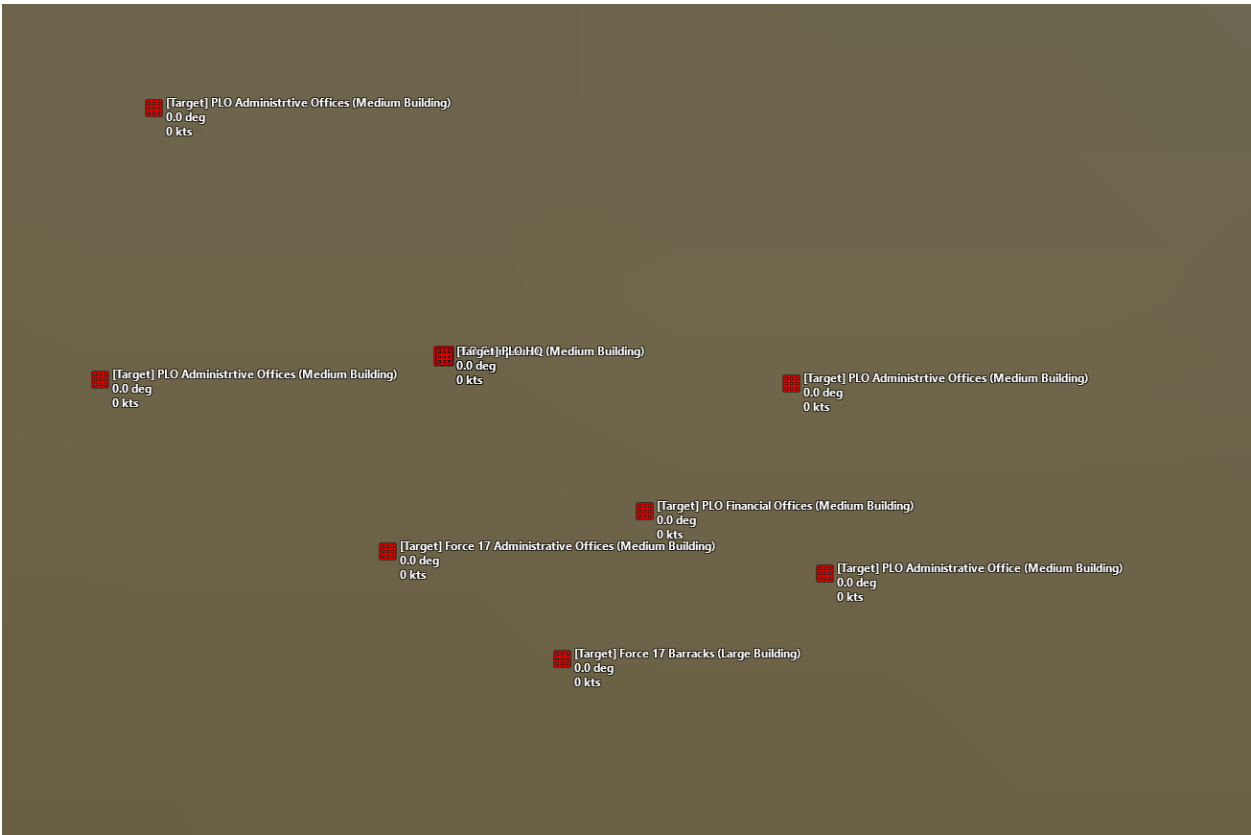
There is a lot of information here, namely that we are using the GBU-15, a very powerful bomb with the ability to glide significant distances as well as use a laser to hit a target. We also can see that we have a 900 nm strike radius without needing to refuel. We can quickly measure the distance on the world map by pressing Control-D and then clicking on the map and dragging a line.

Doing so we can see that the range is a staggering 1300nm! This means that we will have to either refuel on the way over or refuel on the way back. Ideally, we will do both if we have enough fuel in the tankers to do so as it gives us the most operational flexibility.

Pause and unpause the scenario. You will notice that a series of new contacts appear on the screen near Tunis.



We are interested in the ones on the southeast edge of the bay. Press the number 9 on your numpad to switch to individual view if the group view is preventing you from seeing the individual targets.



Looks like a pretty spread out group. If we select any of them we can see that their database entry says they are variations of Large and Medium buildings. Each of these has some light armor protection and a total damage points of about 450 for the large and 300 for the medium. That’s important to know since it tells us about how many GBU-15s we’re going to need to ensure target destruction.

WARHEADS

#299 - Mk84 2000lb GPB [429kg/945lb Tritonal] (643.5 DP)

VALID TARGETS

Surface Ships
Land Structures (Hard)
Land Structures (Soft)
Mobile Units (Hard)
Mobile Units (Soft)
Runways

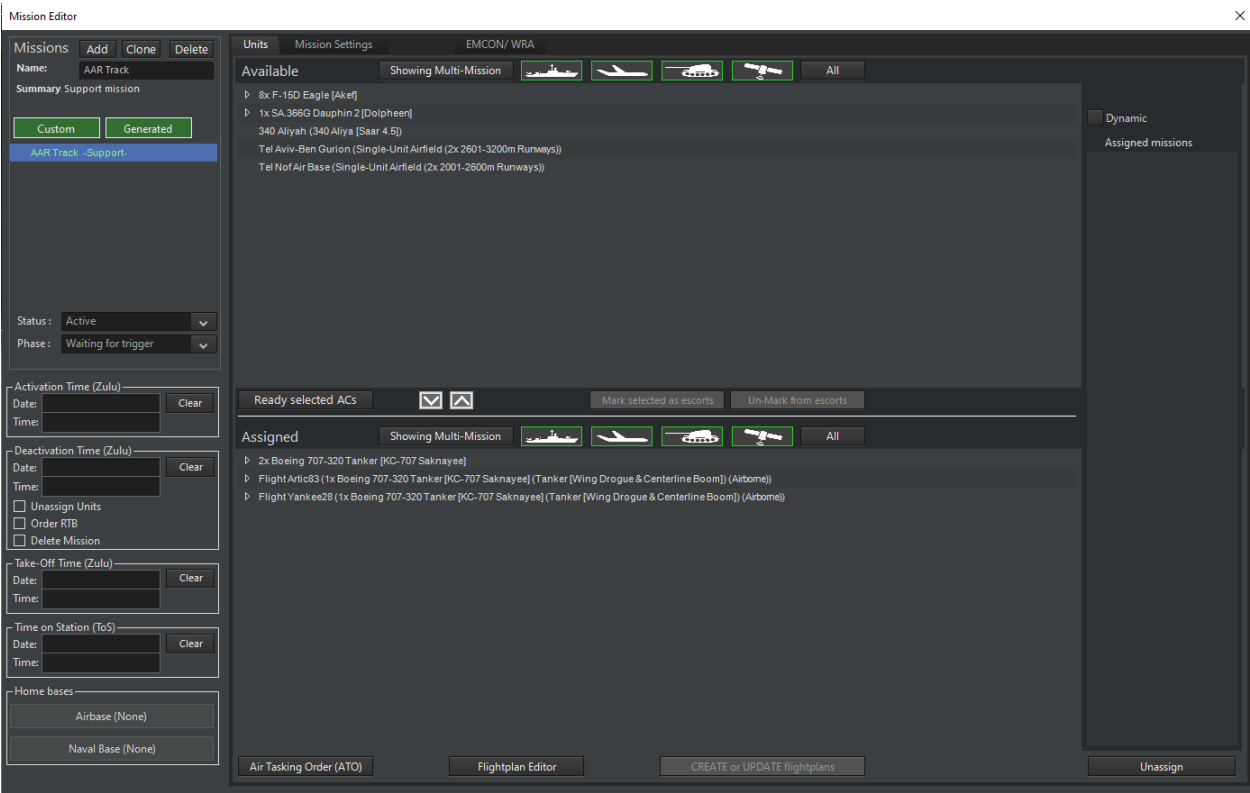
WEAPON RELEASE AUTHORIZATION WPAV DEFAULT

Checking on the database entry for the weapon shows that each of these weapons will be quite capable of a kill, but we only have 6 of them at our disposal since each F-15D can only carry a single one of these weapons. In longer scenarios we could go back and rearm, but in this case we will be cutting it a bit close on scenario time if we try that, so we’ll prioritize the larger buildings and let our sweep aircraft mop up anything that might still be standing with some careful application of Mk82s from the backup aircraft.

10.2.2 Executing our Mission

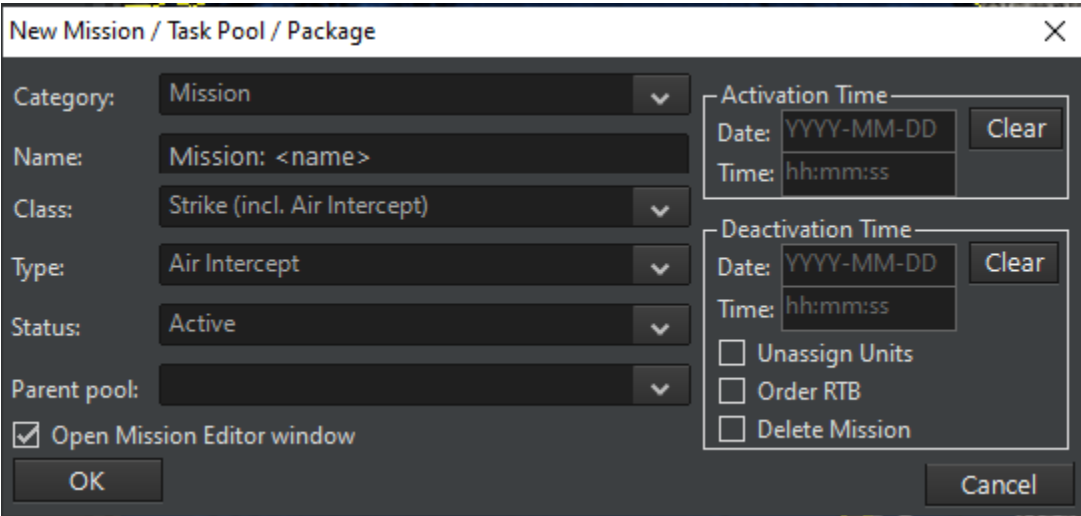
For this scenario, we could potentially fly the entire mission manually, and this could work given how simple our scenario is, but for the purposes of this guide, we will use a strike mission to do so.

Start by bringing up the mission editor by pressing F11 on the keyboard or pressing Mission Editor on the toolbar.



We can see that we already have a refueling flight in position and that there are a few extra tankers available to keep the track operational without worrying about having to chase a tanker back to base with an empty fuel tank.

Select the PLO targets in Tunisia with your left mouse button and then click Add next to Missions.



We will make a few changes here and give it a name.

New Mission / Task Pool / Package

×

Category: Mission

Name: Strike with LGB

Class: Strike (incl. Air Intercept)

Type: Land Strike

Status: Active

Parent pool:

☒ Open Mission Editor window

OK

Activation Time

Date: YYYY-MM-DD
Clear

Time: hh:mm:ss

Deactivation Time

Date: YYYY-MM-DD
Clear

Time: hh:mm:ss

☐ Unassign Units
☐ Order RTB
☐ Delete Mission

Cancel

We will be using two missions for this operation, one will be the primary strike and the other will be the sweepers to do BDA and clean up any remaining targets. While it is possible to put the entire group into one mission, this will give us interesting flight size issues later.

Press OK and head to the next page.

Mission Editor

×

Missions

Add Clone Delete

Name: Strike with LGB

Summary Land Strike

Custom Generated

AAR Track -Support-

Strike with LGB -Strike- [8 Targets]

Status: Active

Phase: Waiting for trigger

Activation Time (Zulu)

Date:
Time:
Clear

Deactivation Time (Zulu)

Date:
Time:
Clear

☐ Unassign Units
☐ Order RTB
☐ Delete Mission

Take-Off Time (Zulu)

Date:
Time:
Clear

Time on Target (ToT)

Date:
Time:
Clear

Home bases

Airbase (None)

Naval Base (None)

Units

Mission Settings

8 Targets

EMCON/ WRA

Available

Showing Multi-Mission

4x Boeing 707-320 Tanker [KC-707 Saknayee]
8x F-15D Eagle [Akef]
1x SA.366G Dauphin 2 [Dolpheer]
340 Aliyah (340 Aliya [Saar 4.5])
Tel Aviv-Ben Gurion (Single-Unit Airfield (2x 2601-3200m Runways))
Tel Nof Air Base (Single-Unit Airfield (2x 2001-2600m Runways))

Dynamic

Assigned missions

Ready selected ACs

☒
☒

Mark selected as escorts

Un-Mark from escorts

Assigned

Showing Multi-Mission

☒
☒
☒
☒

Air Tasking Order (ATO)

Flightplan Editor

CREATE or UPDATE flightplans

Unassign

Before we add any units to the mission, we'll need to set up a few things in Mission Settings first:

Mission Editor

Missions Add Clone Delete

Name: Strike with LGB

Summary Land Strike

Custom Generated

AAR Track -Support-

Strike with LGB -Strike- (8 Targets)

Status: Active

Phase: Waiting for trigger

Activation Time (Zulu)

Date: Time: Clear

Deactivation Time (Zulu)

Date: Time: Clear

☐ Unassign Units

☐ Order RTB

☐ Delete Mission

Take-Off Time (Zulu)

Date: Time: Clear

Time on Target (ToT)

Date: Time: Clear

Home bases

Airbase (None)

Naval Base (None)

Units **Mission Settings** 8 Targets EMCON/ WRA

Strike / Air Intercept Escorts

Mission triggers when contact is minimum: Unknown

Aircraft settings Ship / submarine settings

Flight size: 3 ☒ Enforce flight size (by base, AC type and loadout) Fill empty slots

Minimum # of ready strike a/c required to trigger mission: No preferences

Maximum # of flights allowed to fly mission: No preferences

Tankers (AAR): Inherited, Allow, but not tankers refueling tankers Configure

Fuel / ordnance: Expend/jettison or bring back A/G ordnance as per loadout setting

Radar usage: Use mission EMCON for whole flightplan

Minimum strike radius (distance to target): 0 nm

Maximum strike radius (distance to target): 0 nm

Cruise formation:

Attack formation:

Time-of-day:

Weather:

Attack Method: Single-axis stacked ToT (Trail Attack), independent aim, 30 second separation (typical)

Split Distance: 10nm (compressed)

☐ Allow off-axis attack

☐ One time only (auto-generation by Mission AI)

☐ Use pre-generated flightplans only (no auto-generation by Mission AI)

☒ Include in Air Tasking Order (ATO)

Air Tasking Order Flightplan Editor CREATE or UPDATE flightplans

We are shrinking the flight size to 3 since we have 6 strikers available to us. We are also setting the attack method to a stacked independent attack with a tight split to make the attack happen all at once, reducing the time our opponent will have to scramble their forces. Now click on Configure next to refueling.

Tanker Planner

☒ Use nearest tanker with enough fuel to serve

☐ Use tankers assigned to specific missions

AAR Track

Mission planning details

Minimum number of tankers needed: Not specified

Minimum number of tankers airborne: Not specified

Minimum number of tankers on station: Not specified

☐ Launch mission without tankers in place (extremely risky!)

Mission execution details

Maximum number of receivers in queue per tanker: Not specified

Receivers start looking for tanker when down to 50 percent of mission

☐ Tanker follows receiver's flightplan (enroute refueling)

Airborne receivers can book tankers within...

☐ Receiver's tactical range with current fuel load and flight profile

☒ 500

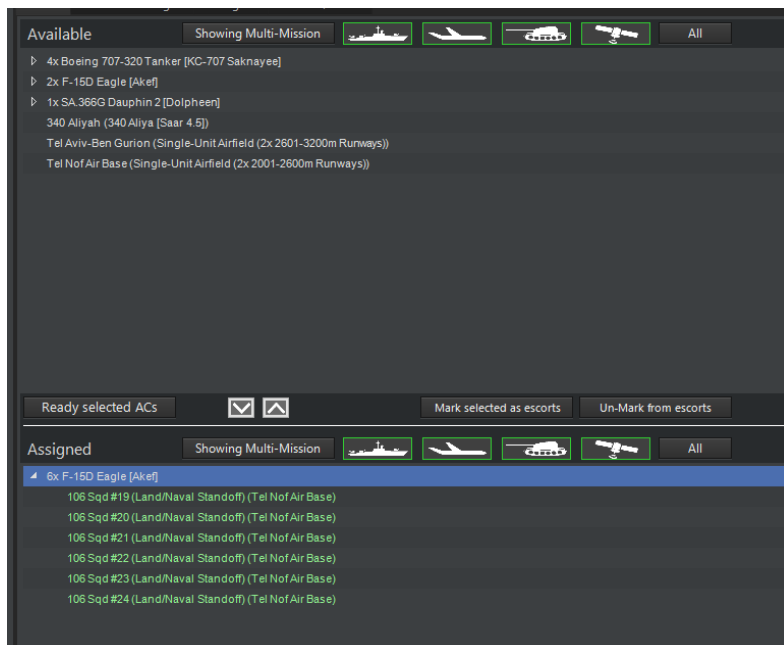
☐ 250

☐ 100

☐ 50

We will define it so that we're using tankers from the AAR Track mission, to look for fuel at 35 percent of fuel and to NOT allow enroute refueling. If we opt for this, our tankers will run dry on their way back to the refueling track leaving us in a tough spot logistically.

Now return to the units page and add the 6 F-15Ds with the LGBs on board.



Looks good. Head to the targets tab and set Opportunity Scrambling to “Focus on strike targets”, set engage to shotgun to ‘Engage until winchester’ and then set Target Destroyed to “RTB”. These options ensure we won’t get distracted along the way with a tasty target. Finally, click EMCON/WRA. The general WRA is good for this mission as we have a pretty simple package that we are using. We do have a weapon release concern however. If we open up “Weapon Release Authorization” we will notice that the default number of GBU-15 weapons to use against a land structure is either 2 or the amount the missile defense value of that target.

Ship - Merchant / Civilian, 95000+ tons	Inherited, Target's Missile ...	▼
Ship - Surfaced Submarine	Inherited, 2 rnds	▼
Land Contact - Unknown Type	Inherited, 2 rnds	▼
Land Structure - Soft - Unspecified	Inherited, Target's Missi...	▼
Land Structure - Soft - Building (Surface)	Inherited, Target's Missile ...	▼
Land Structure - Soft - Building (Reveted)	Inherited, Target's Missile ...	▼
Land Structure - Soft - Structure (Open)	Inherited, Target's Missile ...	▼
Land Structure - Soft - Structure (Reveted)	Inherited, Target's Missile ...	▼
Land Structure - Soft - Aerostat Moring	Inherited, Target's Missile ...	▼

Unfortunately, if you check the missile defense value of a large building:

GENERAL DATA

Category: Building (Surface) (WRA: Land Structure - Hardened - Building (Surface))
Damage Points: 450
Length: 70 m
Width: 40 m
Area: 2800 m
Combat System: None (Large) (ex: Large Bldg)
OODA cycle: Detection: 45 seconds Observe, Orient, Decide, and Act (real time)
Targeting: 120 seconds (Novice Proficiency Level)
90 seconds (Cadet)
72 seconds (Regular)
60 seconds (Veteran)
48 seconds (Ace)
Evasion: 30 seconds
Missile Defence: 2 Harpoon / SLAM / Maverick equivalents

You'll see that our default WRA will use two of our expensive guided weapons per target! This is not needed or ideal for this scenario, so we'll change the WRA so that it only uses one weapon per target.

Land Contact - Unknown Type	1 rnd		In
Land Structure - Soft - Unspecified	1 rnd		In
Land Structure - Soft - Building (Surface)	Inherited, 1 rnd (Using 'U...)		In
Land Structure - Soft - Building (Reveted)	Inherited, 1 rnd (Using 'U...)		In
Land Structure - Soft - Structure (Open)	Inherited, 1 rnd (Using 'U...)		In
Land Structure - Soft - Structure (Reveted)	Inherited, 1 rnd (Using 'U...)		In
Land Structure - Soft - Aerostat Moring	Inherited, 1 rnd (Using 'U...)		In
Land Structure - Hardened - Unspecified	1 rnd		In
Land Structure - Hardened - Building (Surface)	Inherited, 1 rnd (Using 'U...)		In
Land Structure - Hardened - Building (Reveted)	Inherited, 1 rnd (Using 'U...)		In
Land Structure - Hardened - Building (Bunker)	Inherited, 1 rnd (Using 'U...)		In
Land Structure - Hardened - Building (Underground)	Inherited, 1 rnd (Using 'U...)		In
Land Structure - Hardened - Structure (Open)	Inherited, 1 rnd (Using 'U...)		In
Land Structure - Hardened - Structure (Reveted)	Inherited, 1 rnd (Using 'U...)		In

Looks good. Lastly we'll visit the targeting priority page and prioritize the large buildings over the medium buildings. To do so, we'll simply create a few quick rules:

Units Mission Settings 8 Targets EMCON/WRA			
General EMCON Settings Weapon Release Authorization (WRA) Withdraw & Redeploy (Ships / Subs / Land) Targeting Priority			
Platform	Category	Unit Type	Engagement Timing
Facility	Building (Surface)	Building (Large)	Immediate
Facility	Building (Surface)	Building (Medium)	Immediate
Any	Any	Any	Default

There we go. Now we are going to hit the biggest targets first with the actual amount of ordnance we need to accomplish this mission. Keep in mind we may deal with duds or return fire, but for this scenario, we might as well be efficient.

Return back to mission settings and set a takeoff time of 5 minutes from now.

☐ Unassign Units

☐ Order RTB

☐ Delete Mission

Take-Off Time (Zulu)

Date:1985-10-01

Time:05:05:00

Clear

Time on Target (ToT)

Date:

Clear

We'll do this since we need to coordinate with our sweepers and have them come out a few minutes later. At this stage we could create detailed flight plans including a refueling stop along the way, but for simplicity, we'll manually order up the refueling for the outbound leg.

Now that we're done with that mission, we can set up our sweepers. There are a few different approaches to this that can include doing it manually, creating a patrol mission or simply adding another strike mission.

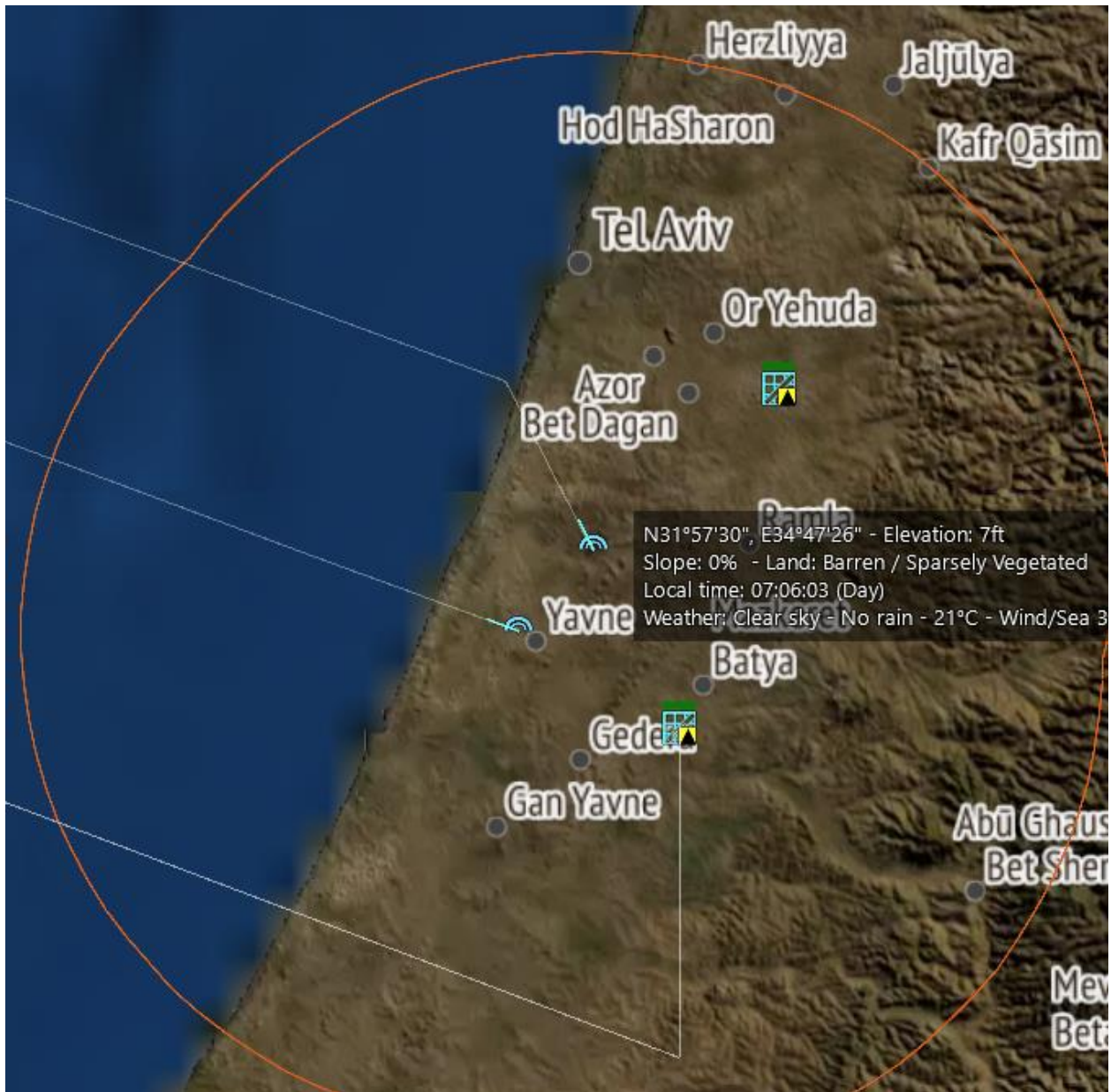
We will:

- 1. Clone previous strike mission
- 2. Change the flight size in mission settings to 2
- 3. Change the WRA so that Mk82s will drop a full stick but from only one shooter
- 4. Set the takeoff time to 2 minutes *after* the previous mission

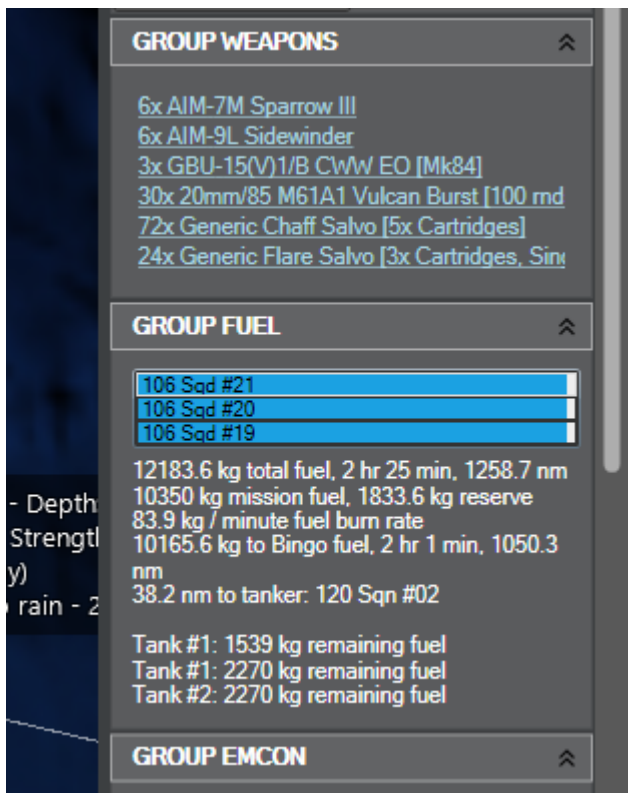
You might think that we should assign a single Mk82 per target, but remember these weapons are unguided and our aircraft will be in the danger zone of low altitude in order to employ them, so we want to limit their exposure as much as possible.

Ship - Surfaced Submarine	Inherited, All weapons	Inherited, Fire weapons fr...
Land Contact - Unknown Type	Inherited, All weapons	1 unit
Land Structure - Soft - Unspecified	Inherited, All weapons	1 unit
Land Structure - Soft - Building (Surface)	Inherited, All weapons (U...	Inherited, 1 unit (Using 'U...
Land Structure - Soft - Building (Reveted)	Inherited, All weapons (U...	Inherited, 1 unit (Using 'U...
Land Structure - Soft - Structure (Open)	Inherited, All weapons (U...	Inherited, 1 unit (Using 'U...
Land Structure - Soft - Structure (Reveted)	Inherited, All weapons (U...	Inherited, 1 unit (Using 'U...
Land Structure - Soft - Aerostat Moring	Inherited, All weapons (U...	Inherited, 1 unit (Using 'U...
Land Structure - Hardened - Unspecified	Inherited, All weapons	1 unit
Land Structure - Hardened - Building (Surface)	Inherited, All weapons (U...	Inherited, 1 unit (Using 'U...
Land Structure - Hardened - Building (Reveted)	Inherited, All weapons (U...	Inherited, 1 unit (Using 'U...
Land Structure - Hardened - Building (Bunker)	Inherited, All weapons (U...	Inherited, 1 unit (Using 'U...
Land Structure - Hardened - Building (Underground)	Inherited, All weapons (U...	Inherited, 1 unit (Using 'U...
Land Structure - Hardened - Structure (Open)	Inherited, All weapons (U...	Inherited, 1 unit (Using 'U...
Land Structure - Hardened - Structure (Reveted)	Inherited, All weapons (U...	Inherited, 1 unit (Using 'U...
Runway Facility - Unspecified	Inherited, All weapons	1 unit
Runway	Inherited, All weapons (U...	Inherited, 1 unit (Using 'U...
Runway-Grade Taxiway	Inherited, All weapons (U...	Inherited, 1 unit (Using 'U...

We are ready to go! Now we can unpause and fast forward to about 5:05 and watch our strikers take to the air on the way to their assigned mission.



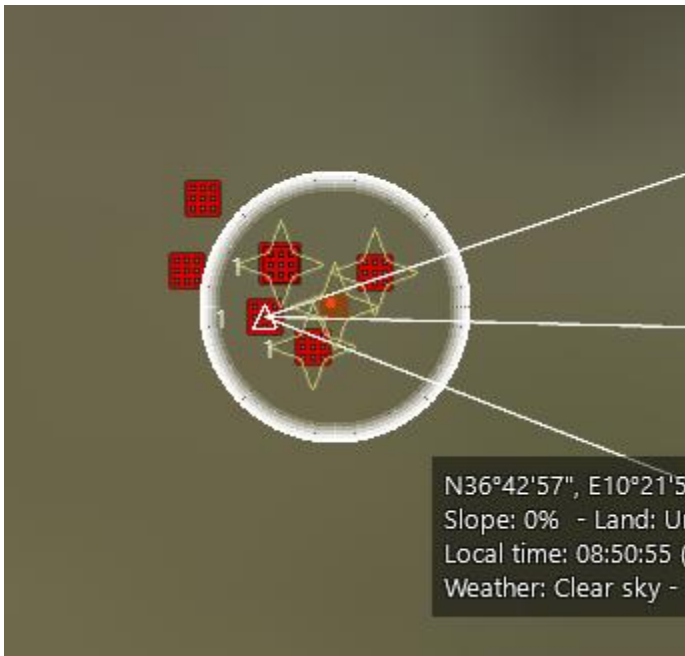
We'll fast forward time for a little bit until our strikers get close to the refueling track. Then right click on each of them and hit "Refuel if possible" Then "Select tankers automatically".



All gassed up and ready to go. We have an interesting problem to consider at this point. Since our tankers will only have enough fuel for 3 hours at this point *and* our strikers have several hours of flight time ahead of them, we are in a position where our tankers might have to RTB **before** our F-15s get back to refuel for them. What we will have to do is manually order our tankers to RTB after all our strikers have fueled up so that a fresh set of tankers are right on station as the F-15s head home. They may have enough fuel to do it without the refueling, but we should be considering this possibility. Now we get to the fun part.

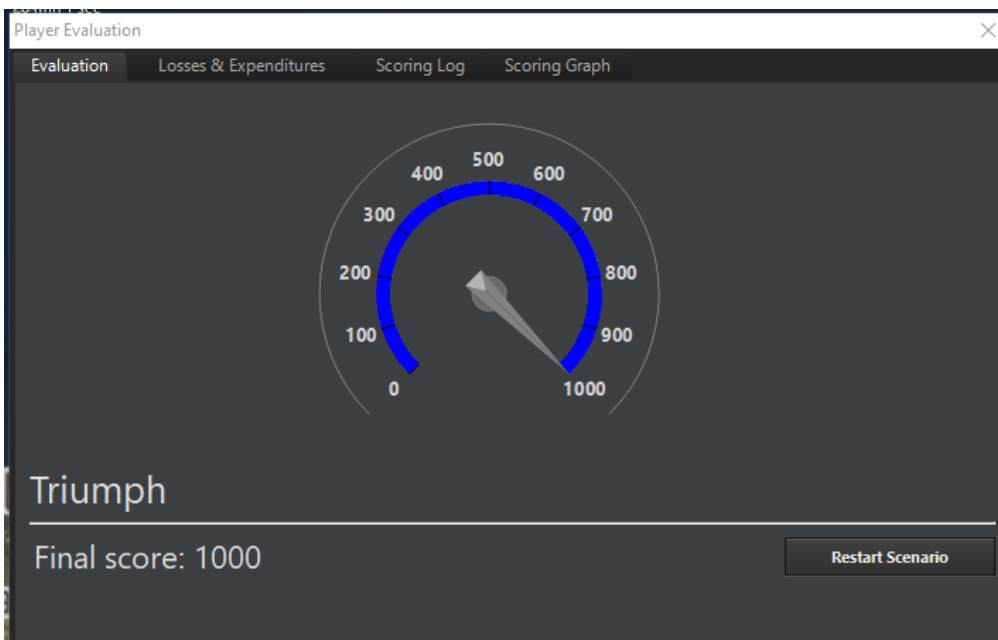


Turning on our radars seems tempting here, but if we do so we could cue anyone around that we're coming. If there was a more robust EW environment in this scenario, we would more or less have to, or at least come in low over the mountains.



LGBs away!

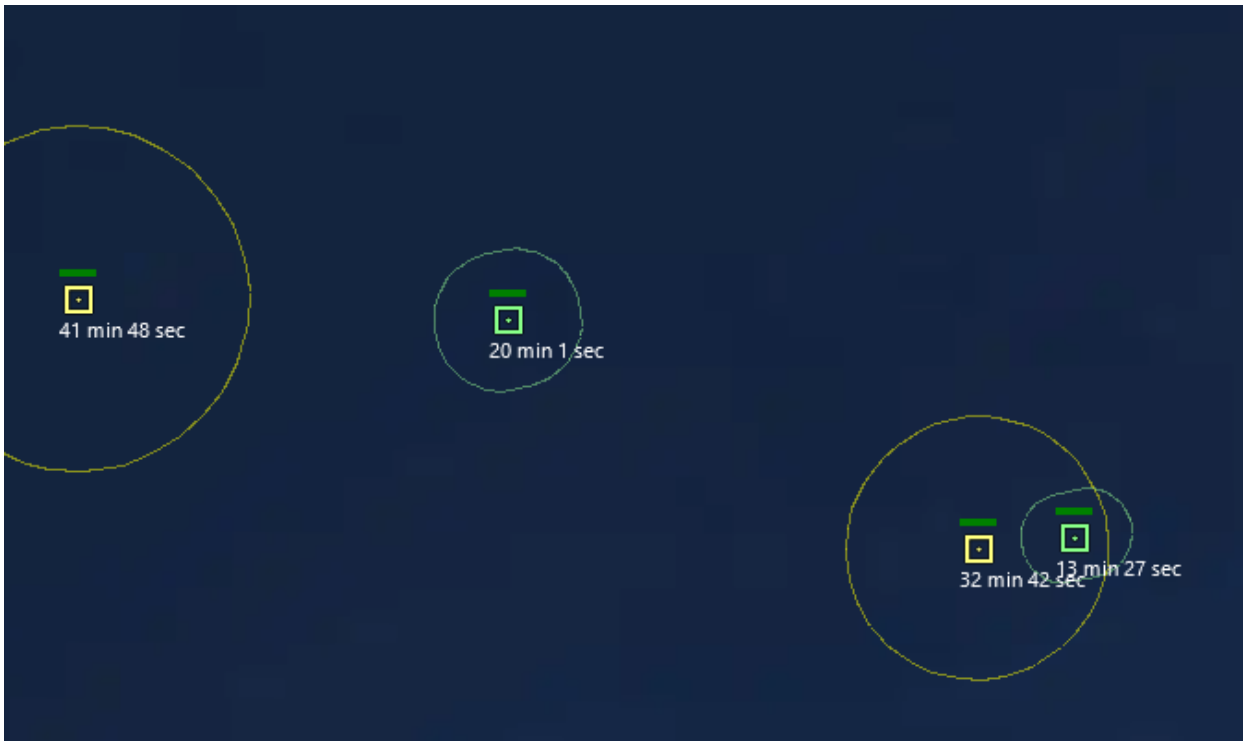
At this point our strikers turn back, refuel as needed and return to base. Once every aircraft has landed, the option to end the scenario will pop up and we can take a look at our performance.



10.2.3 Debrief

As we can see, while the mission was relatively simple, there were many little things that contributed to the overall success. We were fortunate that there were no enemy fighters that came to tango with our strikers, and we were also lucky that there were no SAMs or EW radars. The refueling aspect of this mission was one of the hardest parts to get right and it can surprise new players just how tough it can be to organize a long range strike without doing everything manually.

As you were playing the scenario, you might have noticed some unknown contacts in the water along the way:



These were civilian vessels that were operating in the sea while you were busy navigating 1200 nautical miles. Some scenarios feature very robust civil traffic that make anti-shipping operations quite a challenge.

10.3 Battle of Chumonchin Chan, 1950

10.3.1 Collecting Intelligence and Devising our Strategy

This scenario is a short ASuW operation where we will be utilizing three surface vessels and their gun armament to protect ourselves from waves of irregular forces in the form of PT boats and armed civilian junks. Although there are no missiles in this scenario, much of the surface strategy involved is applicable in the later ages and generations of naval vessel technology.

After starting the scenario, we are presented with the Situation as well as our objectives.

Available Sides: United Nations ▼

Briefing for selected side:

Situation

A North Korean convoy of at least 10 ammunition transports and 4 MTB's and PC craft has arrived near the port of Chumonchin Chan.

Enemy Forces

North Korea

In the Chumonchin area there are known to be North Korean light to medium shore batteries in addition to at least 4 MTB's and PC.

Surface

- Civilian Junk [35m, Armed]
- Civilian Junk [35m]
- TK P-4 [Pr.123K]

Friendly Forces

The Allied flotilla in the area of operations consists of USS Juneau, HMS Black Swan and HMS Jamaica.

Surface

- C 44 Jamaica
- F 57 Black Swan
- CLAA 119 Juneau

Mission

Patrol the Korean coast near Kangnung and engage any North Korean transports or warships.

Execution

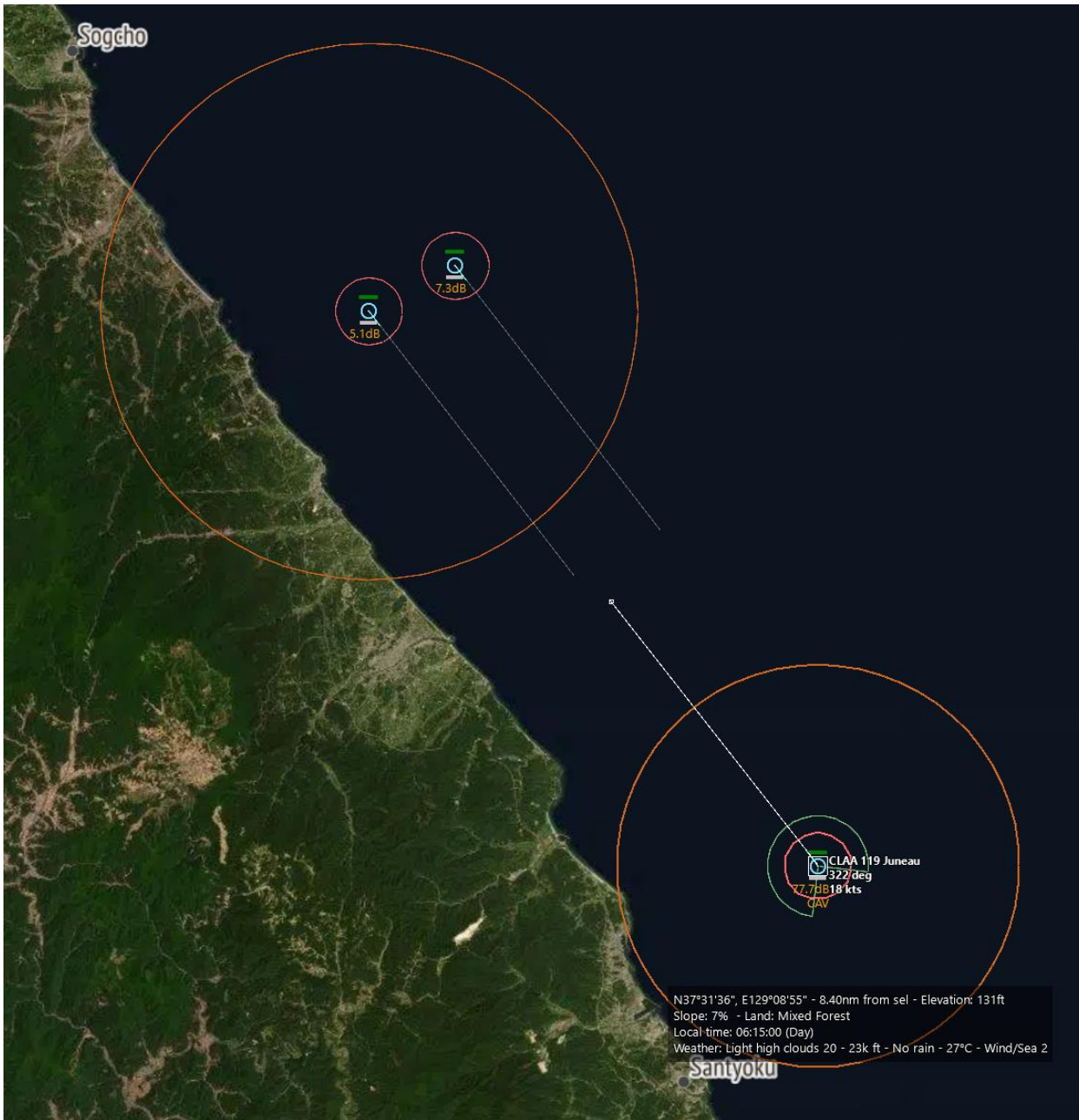
1. Conduct a full sensor sweep of the area
2. Sink a minimum of 6 transports in the convoy
3. Sink a minimum 3 of the escorting MTBs and PC craft

Enter scenario

There are several key points:

1. We are dealing with an non-symmetrical force meaning we will not be doing a rolling gun battle with other vessels of the same capabilities
2. Our forces are mixed and older, but still very capable designs
3. Our orders are to conduct a sensor sweep of the area as well as sink as many enemy vessels that are either torpedo boats or junks.
4. Our previous mission was to shell shore batteries, but that is not required for this scenario

Upon entering the scenario, we can see the base layout of our forces:



The AA cruiser is to the south and the light cruiser and frigate are to the north. The two formations are roughly 26 nm apart meaning at flank speed, the best our solo ship can hope is to meet up with the main formation in about 49 minutes, which is a very long time if you are dealing with light craft. The northern vessels are about 4 nm apart making mutual support a little tricky, at least initially.

Looking at the scenario platforms we can find out a few different and useful things, namely the enemy will have TK P-4s and light junks. The P-4s are the real threat as they pack the 43-36N torpedo with a maximum range of 3 nm.

SENSORS / EW					
Model	Max Range	Notes	Abilities		
1x Skin Head [Zarnitsa]	18 nm	Radar, Air & Surface Search, 2D Short-Range	Early 1950s Technology, Air Search, Surface Search, Range Information, Speed Information, Heading Information, Operating bands (search & track): E		
1x Mk1 Eyeball	15 / 50 nm		Air Search, Surface Search, Ground Search (Fixed), Ground Search (Mobile), Range Information, Heading Information		

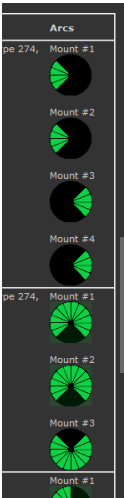
MOUNTS / STORES / WEAPONS					
Mounts (Guns/Launchers/Ejectors/etc.)					
Mount	Capacity	Launch Interval	Armor	Onboard Sensors	Weapons (per mount)
1x 14.5mm/73 Twin	15	1	None		15x 14.5mm/73 Twin Burst [20 rnds] Gun. Targets: Aircraft, Helicopters, Surface Ships, Land Structures (Soft), Mobile Units (Soft). Max Range: 1 nm. Warhead: 14.5mm HE. Max Rate of Fire: 1000 rpm.
2x 450mm Single TT	1	300	None		1x 45-36N Torpedo. Targets: Surface Ships. Max Speed: 40 kts. Max Range: 3 nm. Warhead: 250kg HE Torpedo.

Magazines					
-----------	--	--	--	--	--

Fortunately the torpedoes are straight runners meaning that we could thread a torpedo spread if we are made aware of it. Other information we can determine is that the torpedo boats have a top speed of 50 knots and that they have a Skin Head [Zarnitsa] radar making it easier to identify them from their emissions if we need to establish who to prioritize.

The Junks have a top speed of 18 knots and have some very basic armament consisting of RPGs and 12.7mm machine guns. They would only be a threat to us if they were allowed to get within shouting distance, so we know who to prioritize when the shooting starts.

Our forces are a little different. The C 44 Jamaica is a light cruiser from World War 2 and it is packing 4" and 6" guns with a maximum range of about 12 nautical miles, giving us quite a stick if we need to pick off the torpedo boats before they can close. The Jamaica has a flank throttle of 31 knots and importantly, we know the orientation of the gun batteries from "Arcs" display in the database as seen to the right. If we want to have maximum effectiveness, we'll need to either face the target directly, or we'll need to be firing off the port or starboard side. This also means that in a chase situation, we are at a *major* disadvantage since we only have one turret of 6" guns facing rearward.



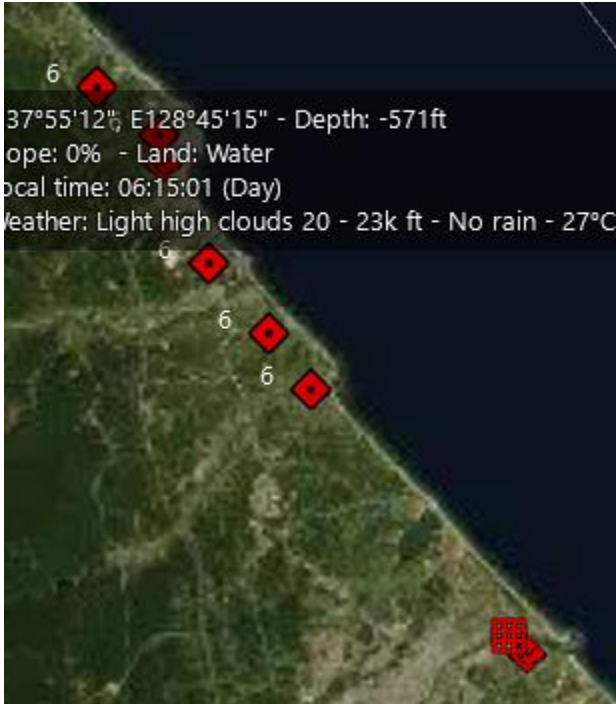
The F 57 Black Swan is an older frigate packing 4" guns and with a much slower flank speed of 21 knots. If we were picking the liability, this would be it since it lacks the ability to concentrate fire like the cruiser does, and the 4" guns, while they have a great rate of fire, only can hit targets 6 nm away. As a point of comparison, the 50 knot torpedo boat can cover 6 nm to 3 nm range in 3 minutes and 36 seconds, leaving us a very tight window if the PTs are able to close.

The CLAA Juneau is one of the best units of the scenario when it comes to firing arcs, as it has 6 5" gun mounts along its length and each of those has a range of 9 nm. Unfortunately since it is literally an hour away from the fight, we'll have to use it to mop up or to cut off the enemy retreat.

10.3.2 Executing the Mission

The key tactic for this scenario will be keeping the enemy as far away as possible from our vessels, keeping our speed up, and making sure as many firing arcs as possible are open for the purposes of engagement. The two northern vessels need to be brought into a position where they can mutually support each other and the southern vessel needs to get to the scene as quickly as possible.

Unpause the scenario and give it a few moments to get the initial sightings, then press pause again.



There are also some shore batteries in the mix that we must stay away from. Checking the database indicates that they have a range of about 4 nautical miles. We could create a non-navigation zone around them if needed, but they are not as big a priority as the incoming attackers.

Select each of your vessels and turn their surface search radars on.

Sensors for: C 44 Jamaica		
☐ Unit obeys EMCON (disables manual sensor control)		
Quick selection: ☒ Radars ☐ Sonars ☐ Offensive ECM		
Sensor	Sensor Type	Active
Type 274	Radar, FCR, Weapon Director	☒
Type 277P HF	Radar, Height-Finder, Medium-Range	☒
Type 277Q	Radar, Air Search, 3D Medium-Range	☒
Type 281	Radar, Air Search, 2D Medium-Range	☒
Type 283 GFCR	Radar, FCR, Weapon Director	☒
Type 293Q	Radar, Target Indicator, 2D Surface-to-Air & Surface-t...	☒
FH-4 HF/DF	HF/DF	
Mk1 Eyeball		

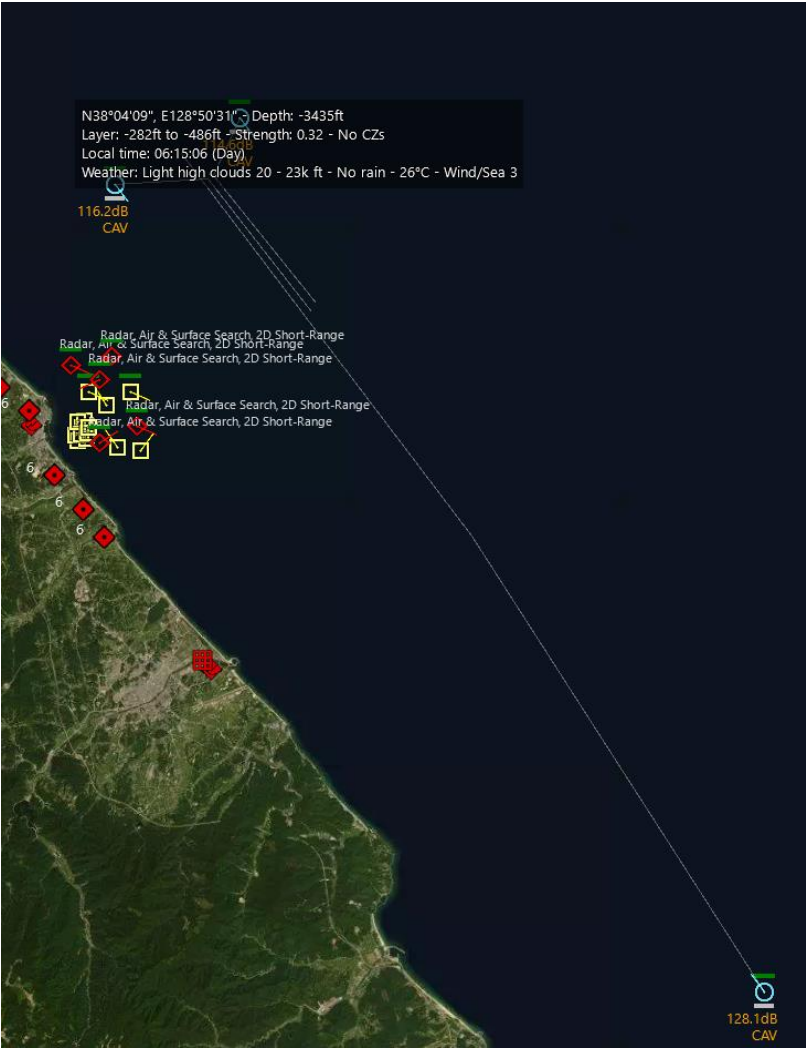
While it is tempting to let one ship keep its radar on while the others do not for the purposes of denying the enemy intelligence, one of our objectives was to do a full sensor sweep. Let's see what our radars uncovered.



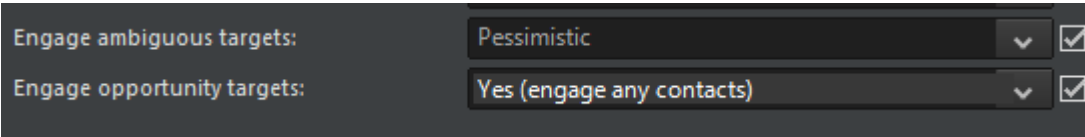
Wow. That's a few ships. Luckily our HF/DF crew on board the vessels have immediately picked out the emissions of several 2D Short-Range radars. From what we know about the units in this scenario, the only ships with that type of radar are the torpedo boats, making it very easy to identify them as hostile. Click each

of the ships with the Air & Surface Search, 2D Short-Range radar and press H to mark them as hostile. You'll notice when you hold your mouse over them, that the ESM crew has already marked them as PT.

Now that the enemy is identified, we can start maneuvers. Get all of your ships up to flank speed and maneuver the Jamaica towards the frigate while sending the Black Swan while bringing the course of the Black Swan towards the cruiser. The Juneau to the south should be plotted to get into the action, and we'll tweak that a little later on depending on what the opponent does.



Our goal here is to beam the enemy forces as best we can while building up enough speed to put some distance between us and the deadly torpedoes. While we're doing this, we'll take a quick visit the Doctrine & ROE for side: United Nations (Control - Shift - F9) and set "Engage Opportunity Targets" to yes.



Now we'll unpause and monitor the action for a minute. The closest enemy to one of our ships is about 5.5 nm away, giving us only a few minutes to destroy the incoming vessels before they get into firing range.

Notice what happens as the battle starts:



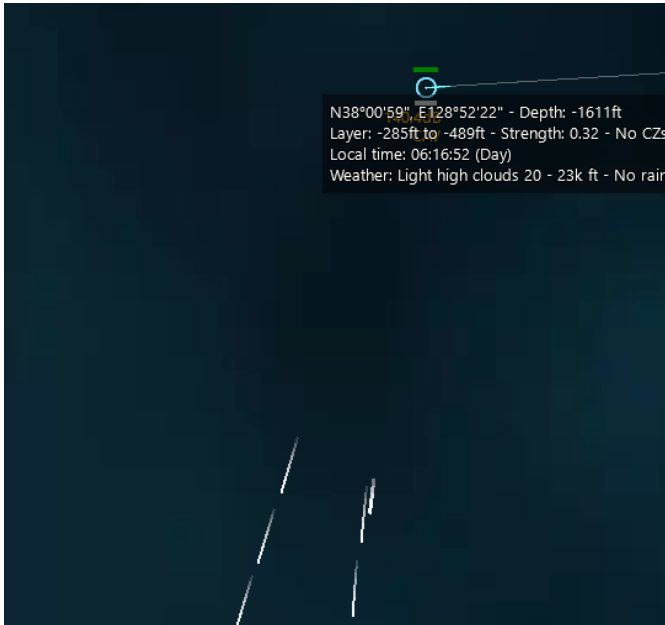
Four other vessels immediately point at us and start raising steam. If we check with our distance measuring tool, we can actually see they are following an intercept course with our vessels.

While this is initially concerning, we can probably assume those vessels are the armed junks and while they are dangerous at *very* close range, we'd rather our cruise prioritize the incoming PTs *before* declaring them hostile and wasting firing effort on something that is less of a threat than the torpedoes that may follow.

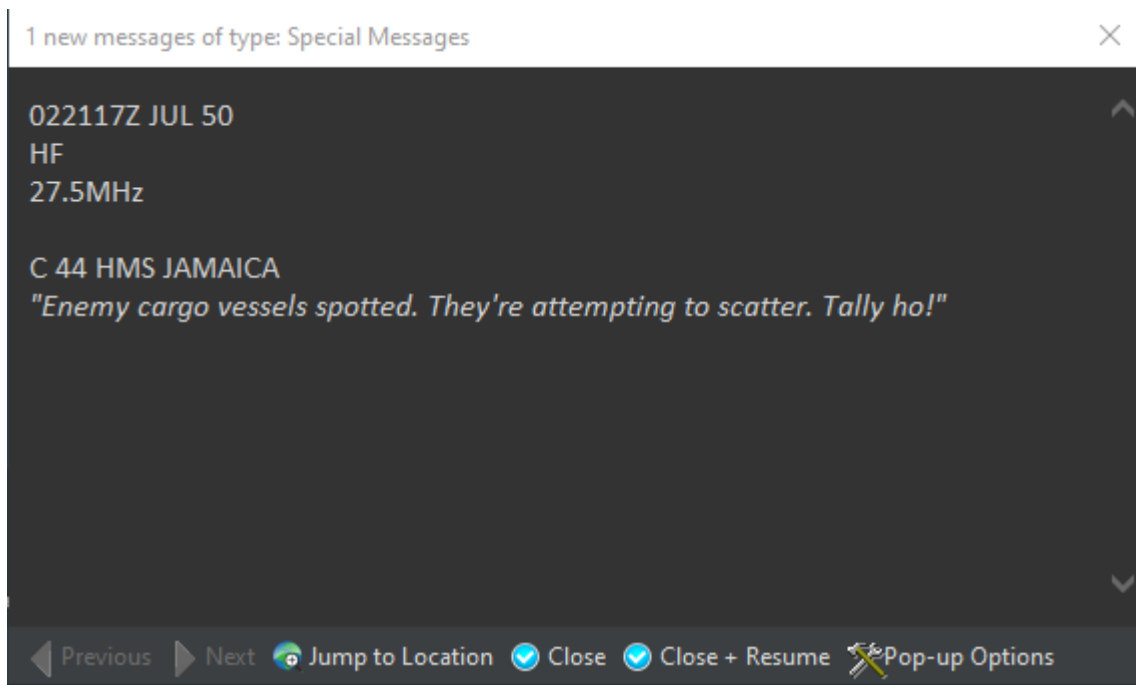


We can see the 6" shells from the cruiser bracketing the incoming PT boat. Even though the shells were not direct hits, the concussion can still do a lot of damage. Making five enemy units combat ineffective is better than destroying one and leaving the other four still ready to go.

It is possible using manual targeting to 'force' the ship to select a new target while shells are on the way, but this can be very difficult to manage in a very pitched naval battle.



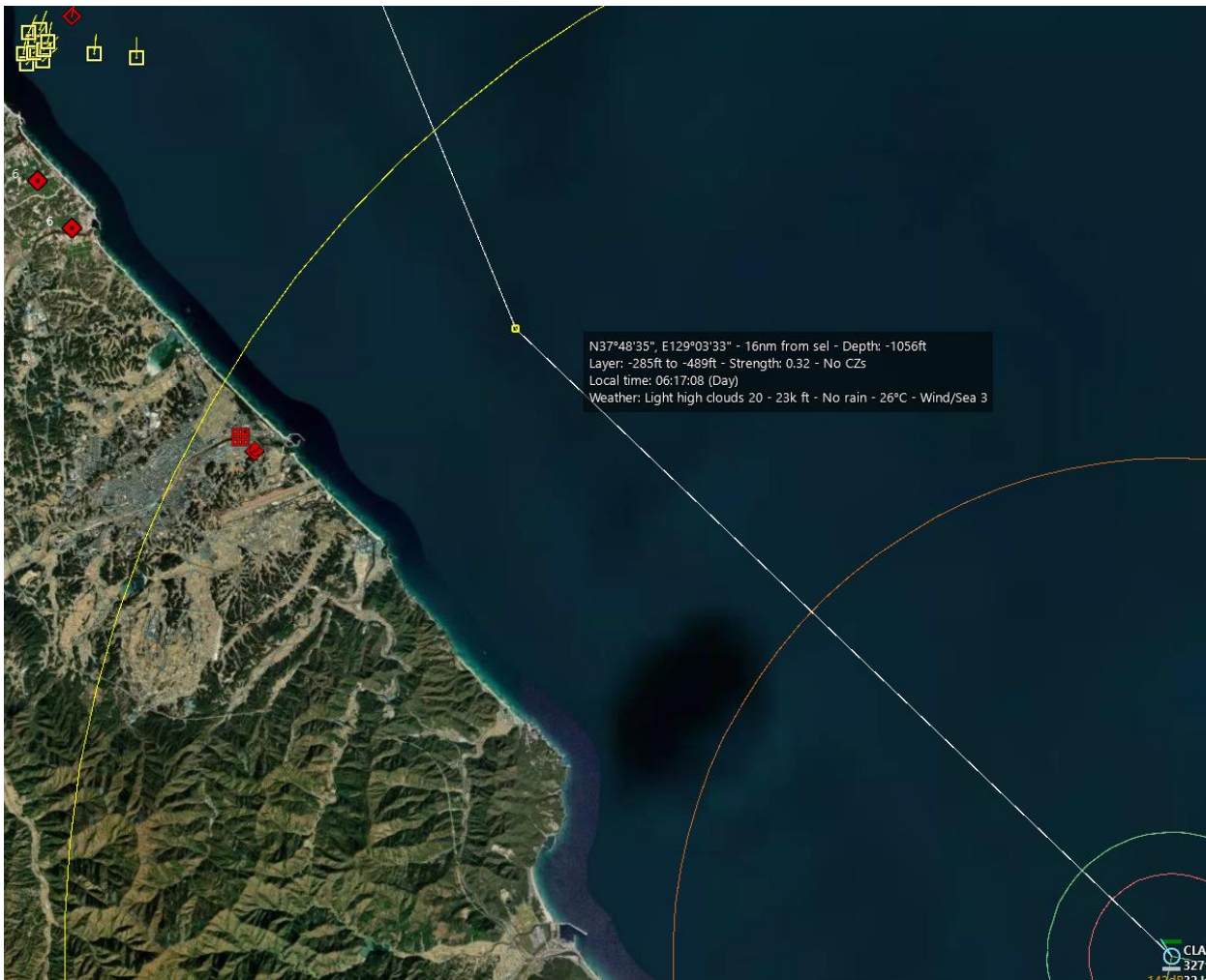
You can see how the cruiser splits its fire automatically since it has multiple director control towers and associated radars regardless of our need to manually target.



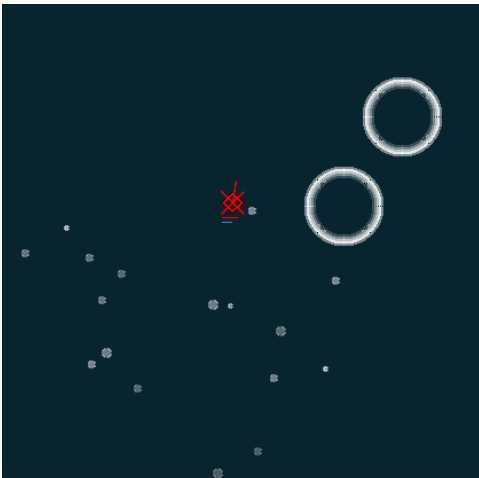
We now know what to look for.



There it is. The enemy convoy has spotted us and is turning to flee to the south. Luckily we have Juneau on the way to block their retreat.



Move the targeted destination slightly southward to cover the distance quicker. But do **not** come within 4 nm of the beach, or you may be attacked by shore batteries.



One PT down, 4 to go. If needed, keep the ships perpendicular to the incoming line of enemy fire. This strategy will keep true all the way into the near future with the advent of high-speed anti-ship missiles.

You may notice the firing rate of the cruise suddenly drops off at this point in the battle. This is not due to battle damage, but rather if we take a quick look at the weapons tab of the Jamaica we can see:

Mount (click weapon for DB info)		Time to fire	Status	R
102mm/45 Mk16 Twin	(20/20)	Reloading: 18 sec	Operational	
102mm/45 Mk16 Twin	(20/20)	Reloading: 18 sec	Operational	
102mm/45 Mk16 Twin	(20/20)	Reloading: 17 sec	Operational	
102mm/45 Mk16 Twin	(20/20)	Reloading: 17 sec	Operational	
152mm/50 Mk23 Triple	(1/1)	Ready	Operational	
152mm/50 Mk23 Triple	(1/1)	Ready	Operational	
152mm/50 Mk23 Triple	(1/1)	Ready	Operational	
40mm/60 Twin Bofors	(42/42)	Ready	Operational	
40mm/60 Twin Bofors	(42/42)	Ready	Operational	

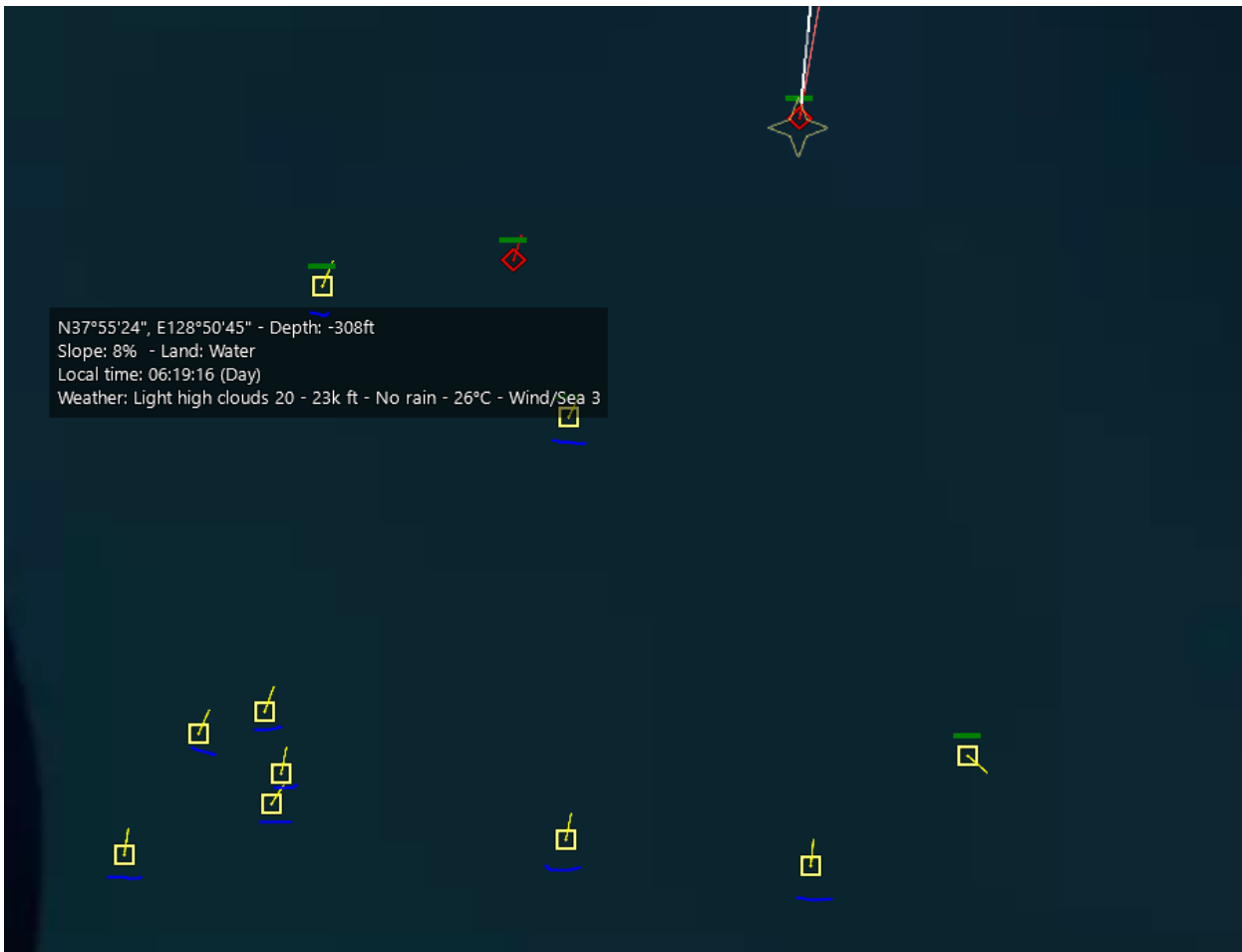
Gun warfare with naval vessels involves a lot of ‘hurry up and wait’, and modern warfare is no different. As the battle continues, you may see something like this occur:



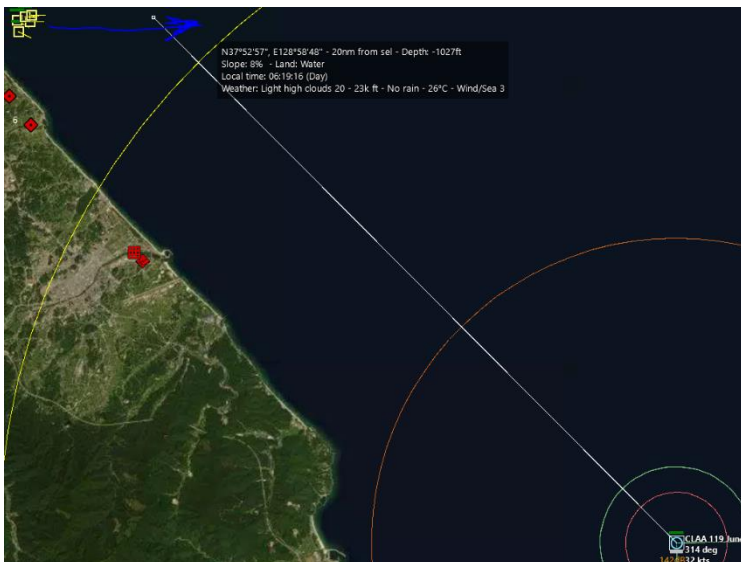
Notice how the Jamaica was busy engaging the other two vessels and missed one coming up the center. A quick check with the range tool tells us that they knew skunk is 4 nautical miles away. Luckily the Black Swan is just coming into range with its own 4” guns and we can click it, press F1 then select the straggler for automatic targeting. It is very tempting to turn the Black Swan perpendicular to the incoming vessel to open up the rear guns to fire, but this would cut into our time needed to close the distance to Jamaica which could become problematic.



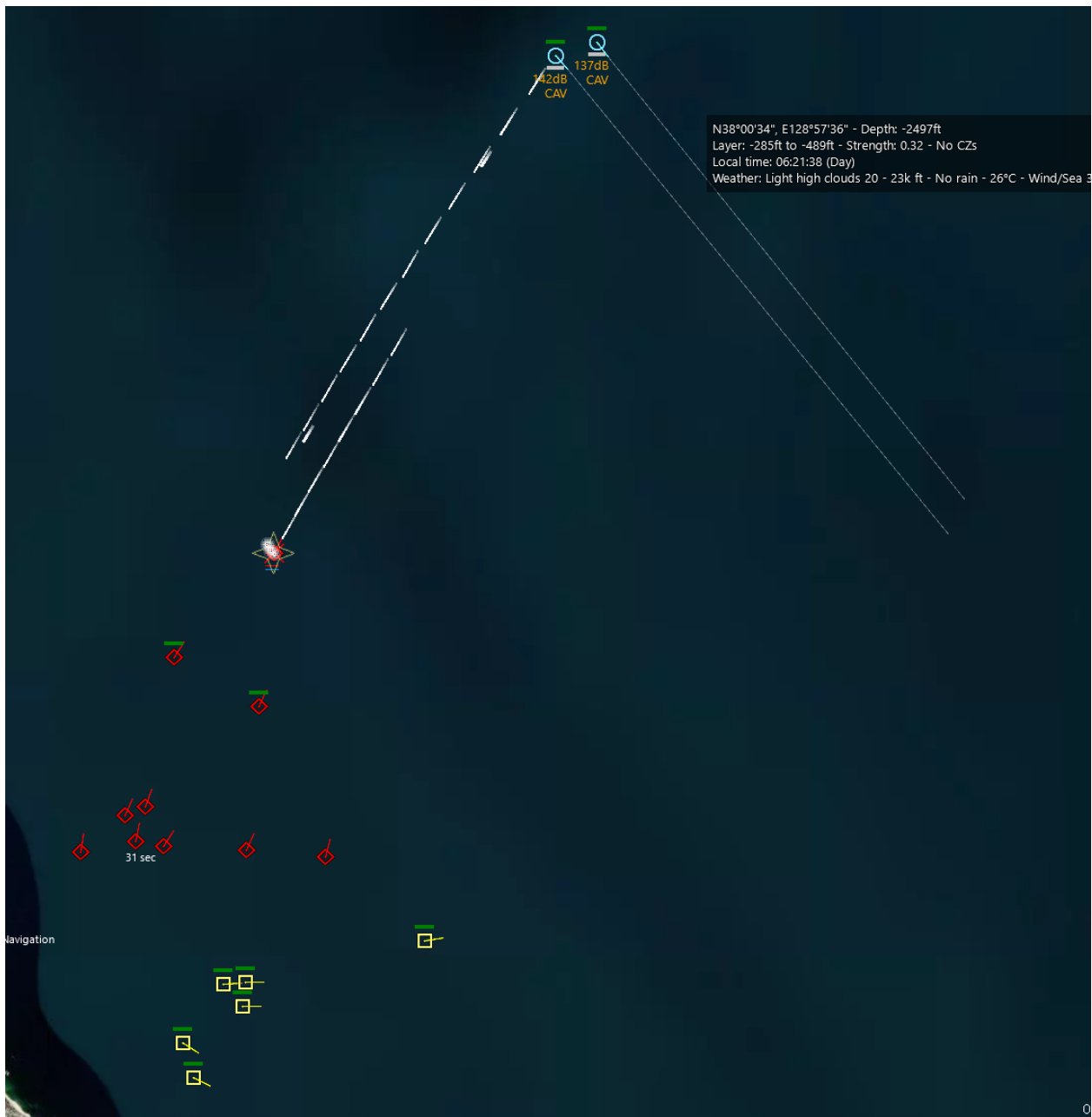
As we discussed in the briefing, mutual support is critical when dealing with gun-based vessels.



Note how all these other vessels turned to intercept us while the group to the south head East. Adjust the course of Juneau to intercept.



You will also notice one vessel that seems to be doing its own thing. This could be an enemy or could be a civilian, and without the presence of any HF/DF on them, we will have to visually identify them later before we close for a kill. Mark the other vessels on an intercept with us as hostile.



Note that both our vessels are working together now, but also observe how those unknown vessels are rushing out to sea. We have to finish up with the lighter enemy vessels first, then we can go for the runners. Remember our southern vessel is steaming up to assist us.



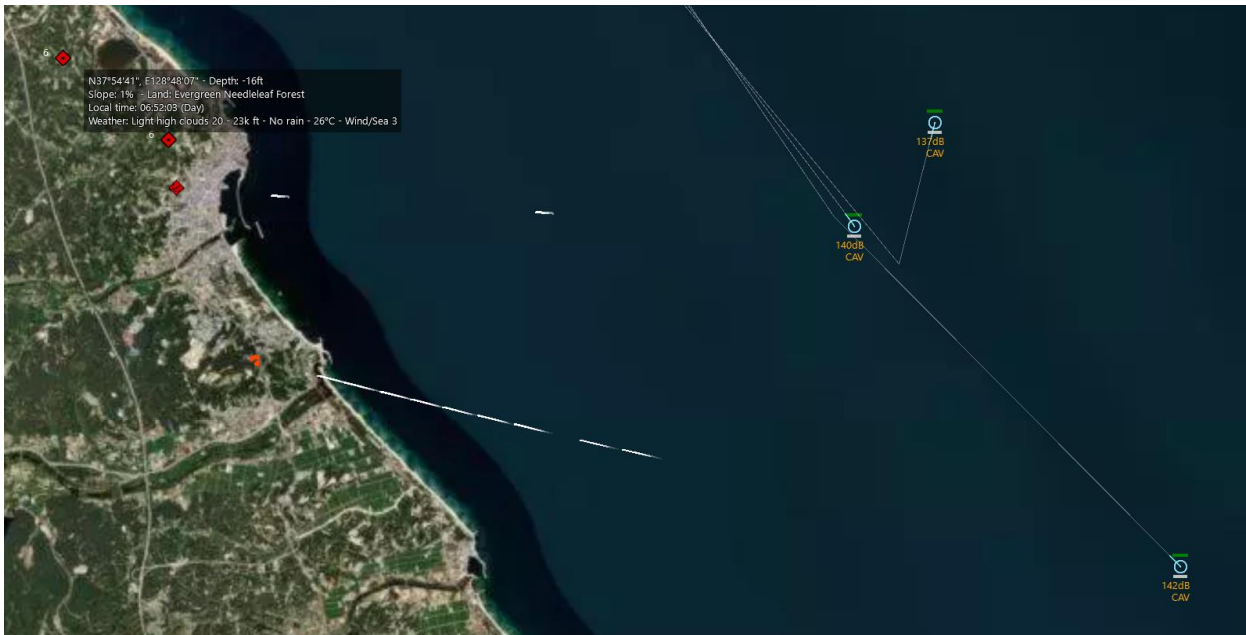
Adjusting our cruise to cut off the retreat. Notice how the western most vessel is slipping out of range, you can tell Jamaica to hit it with the 6" guns or break the Black Swan off to do the deed. Also note that the two vessels are falling out of formation and will need some manual speed adjustments, or to be put in a group to keep them together.



Note that Juneau has started firing on the town marker. We *do not* want that since it will result in a major loss of ammunition for no gain. In COMMAND, objects like cities and towns require much heavier ordnance.

You can quickly shut down the unwanted shelling by right clicking on the town marker and selecting “Filter Out”. This will cause your ships to ignore the target until you reactivate them.

At this point we can either run out the clock to end the scenario or if we’re feeling a little more daring, we can set our ships up to shore the very shore batteries we were originally sent to attack, but this is actually out of the scope of the scenario.



Cleaning up the shore batteries. Remember that these shore batteries have a range of 4 nm and will happily set fire to any vessel that gets too close to shore.

We can use time acceleration to run out the clock and see how we did in the scenario. An interesting point to visit with this run was the Losses & Expenditures page under the “Game” menu.

Losses & Expenditures

Reset All Losses/Expenditures Reset All Side Scores

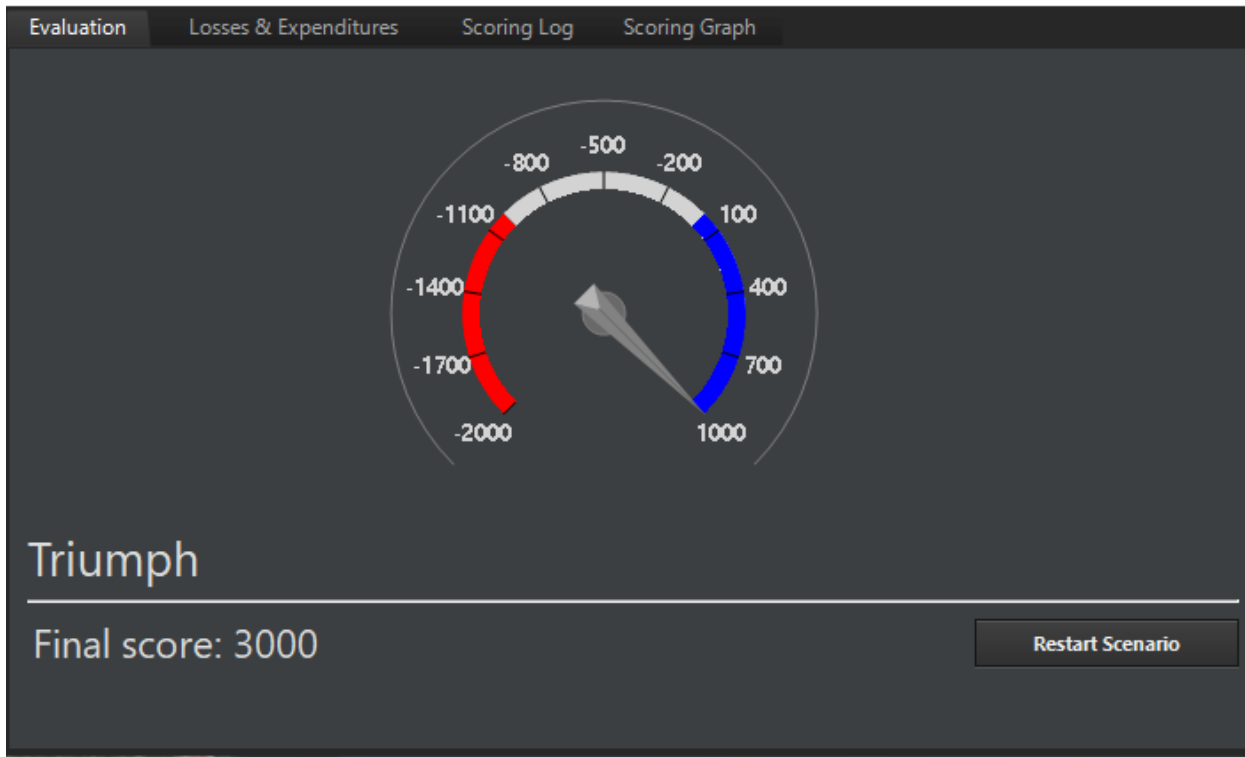
LOSSES:

EXPENDITURES:

1129x 102mm/45 Mk16 Twin HE Burst [2 rnds]
160x 127mm/38 Twin AA-VT Burst [Anti-Aircraft Common, 2 rnds]
332x 127mm/38 Twin HE-PD Burst [HiCap, 2 rnds]
360x 152mm/50 Mk23 Triple CPBC Salvo [3 rnds]
204x 152mm/50 Mk23 Triple HE Salvo [3 rnds]

WEAPONS/STORES LOST:

And of course there’s the final scoring page as well:



10.3.3 Conclusions and Discussion

While the scenario was relatively uncomplicated, it showed many key points in all naval warfare:

- Keep your speed up unless you're dealing with submarines
- Keep your weapons in range and position the most number of your weapons to face the incoming threat
- Keep your enemy out of their effective range
- Use mutual support
- Press the attack and don't let the enemy escape
- Prioritize the biggest threats first
- Monitor your expenditures. In a longer fight, you may have to decide between closing the range to save ammunition and risking battle damage.
- Ships only can target a handful of opponents at the same time. It is possible to overwhelm a ship's defenses with a large quantity of concentrated threats.

The other important takeaway from this scenario is that when missiles get involved in a few years, detectability and detection take a much bigger role in battles since if a surface action group is detected, possibly hundreds of supersonic anti-ship missiles will rain down in a matter of minutes.

10.4 Trapped Under Ice, 1999

10.4.1 Collecting Intelligence and Devising our Strategy

This scenario is a very challenging operation and at the same time an excellent introduction to the challenges inherent in anti-submarine warfare. You may have to retry this scenario several times before achieving the scenario goals as this scenario features random placement of the enemy submarines creating a situation where you may score a lucky win one round or complete destruction on another.

Take a look at the briefing:

Available Sides: United States

Briefing for selected side:

Situation

The Pristina incident has catapulted into a major crisis. Strategic forces on both the US and Russia are increasing their alert status. What few Russian SSBNs are operational are putting out to sea, and those already at their bastion locations are shoring up their escort forces. There is still hope of a diplomatic solution, but that means first demonstrating to the Russians that they have more to lose. Washington has decided to relay a strong message.

Enemy Forces

Russia

Intelligence reports that the first Typhoon-class submarine, the Dimitri Donskoy, is assigned to patrol in this area. The estimated patrol area is marked on your map by the reference points SSBN FZ-1/2/3/4. It will certainly be escorted by at least one and probably more nuclear attack submarines, likely top-line units (Akula / Sierra / Victor II). Because of their strategic duty, these forces are the least affected by the severe cuts in the Russian military. Expect them to have good equipment, and know how to use it.

Submarine

Primary Target

- PLARB-941 Typhoon [Akula]

Potential Escorts

- PLA-671RTMK Victor II [Shchuka]
- PLA-945 Sierra I [Barracuda]
- PLA-971 Akula I [Shchuka-B]

Friendly Forces

Submarine

- SSN 768 Hartford [Improved Los Angeles Class]

Mission

Locate and sink the Typhoon-class submarine.

Execution

1. Destroy the Russian strategic missile submarine known to patrol this area, without fail.
2. Survival of your boat is important but not, repeat not a primary consideration.

We see that we have our work cut out for us. There are three “potential” escorts and each of these is a very capable Russian attack submarine. Our target is the PLARB-941 Typhoon, the largest submarine ever built and was designed from the outset to be able to pop through arctic ice and deliver its deadly load of 20 SS-N-20 nuclear weapons, each with 10 MIRV. Our submarine for this scenario is the SSN 768 Hartford, an Improved Los Angeles Class nuclear attack submarine.

Our goal in this scenario is to destroy the Russian missile sub, even if it costs us our submarine in the process. Destroying the other Russian attack subs is nice, but not needed for this scenario.

As in our other scenarios, we'll carefully browse the scenario units and see if there are any things we can discover about the units that we will be encountering. After looking carefully, there are many things you may discover:

1. The Russian subs primary anti-submarine torpedo is the USET-80K and it has a default range of 5 nm and its seeker is only sensitive out to 1.5 nm. This means our primary anti-torpedo strategy will involve turning and running to get out of its effective cone.
2. The Russian submarines also possess a long range standoff torpedo that is delivered by missile, a weapon that *cannot* be used due to the presence of the ice above our heads. This is the luckiest break we have in this scenario.
3. **All** of the Russian submarines have a towed array sonar system meaning they can hear things both behind it as well as underneath it since the hydrophone is trailed behind the submarine away from its noisy propeller.
4. The Mk48 Mod 5 ADCAP torpedo is our primary weapon and it has a staggering range of 21 nautical miles making it a fantastic ‘recon by fire’ weapon. It also does 270 damage points.
5. The Typhoon is both double hulled and it possesses 75 damage points, which is less than our torpedo is capable of inflicting, but the Typhoon is also well armored meaning it may take more than one clean hit to do the deed.
6. The SSN Hartford is a very capable platform with a top speed of 32 knots and several different types of towed arrays. It is also 10 dB quieter than any of the Russian subs, giving us a huge stealth advantage. This also means that if we detect a Russian sub first, we will almost certainly get the first shot.

7. The patrol area we are covering is 50 nm x 60 nm (30000 square nautical miles of ocean). If we split that into 10 mile wide corridors of search that are 60 nm long, at 5 knots, it would take us 14 hours to complete one sweep. Since we only have 48 hours to complete the mission, this is *not* viable.

As we can see, this is going to be very tricky but it's not impossible. We have a few basic strategies available to us:

1. Conduct an ASW mission in the 'center' of the patrol area
2. Build a search pattern using sprint and drift
3. Drive our submarine into the search area and fire a spread of torpedoes in every direction while listening
4. Run full speed through the search area attracting torpedo attacks and then firing on the bearing of incoming attacks.

When deciding between possibilities, it sometimes helps to create a risk chart:

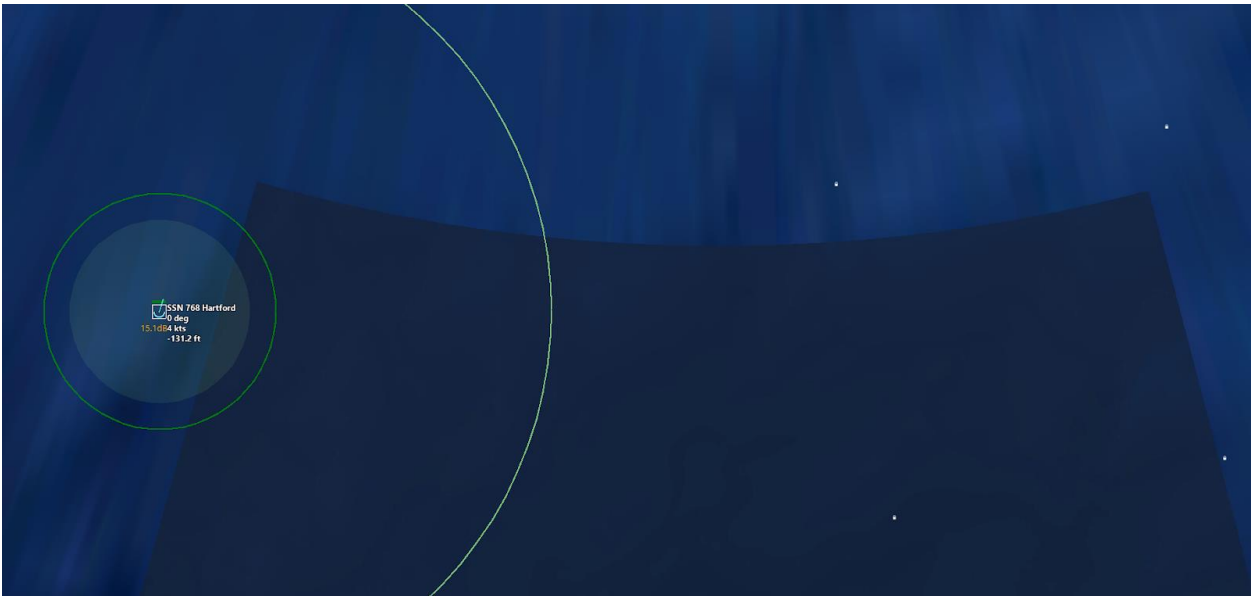
Option	Chance of Success	Risk to us	Complexity
ASW Mission	Low	Low	Low
Sprint and Drift	Medium	Medium	High
Torpedo spread	Low	Low	Low
Full Speed Ahead!	Medium	High	Low

The usual balance is between success and risk. We want the highest chance to succeed, but the one that minimizes losses on our side. Note how strategies that are often lower risk are either *less effective* and/or *less complex*. This is no surprise and it's why there are military colleges dedicated to these subjects.

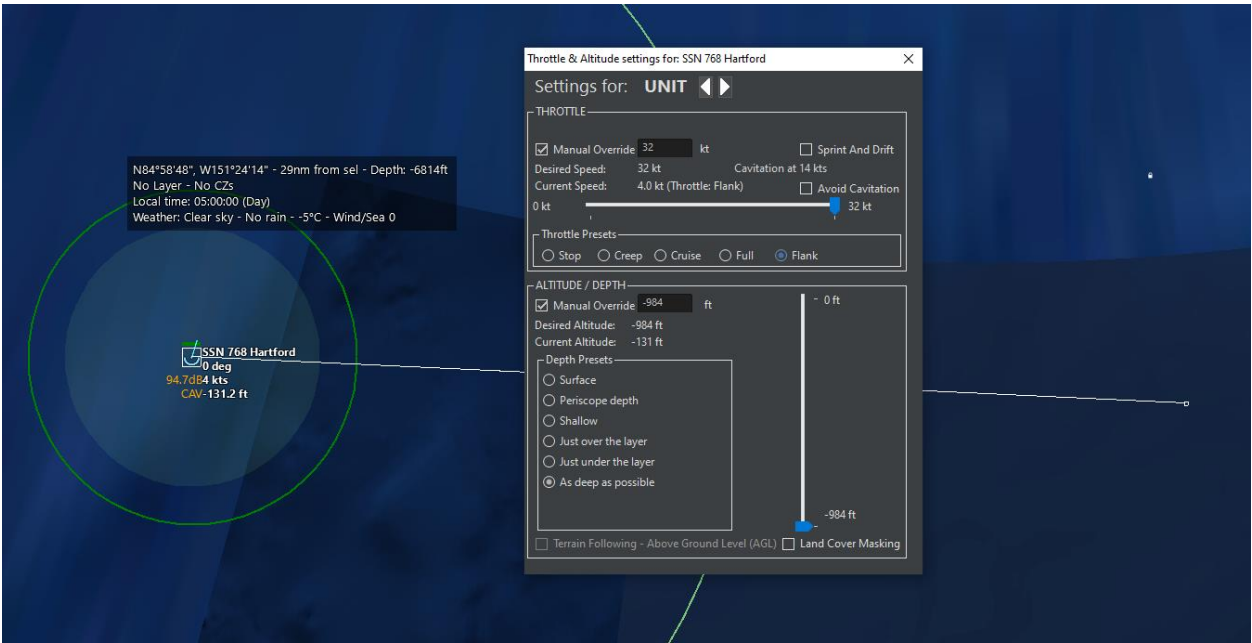
We will opt for two of the strategies, the sprint and drift and the torpedo spread. If we are running out of time in the scenario, we'll engage in a high speed torpedo dodging exercise. Remember as we said in the introduction, even if we get *everything perfect* we may still fail the scenario. ASW is as much about art as it is science, and in other ASW style scenarios, even with the presence of more assets and powerful passive tools such as sonobuoys.

10.4.2 Executing the Mission

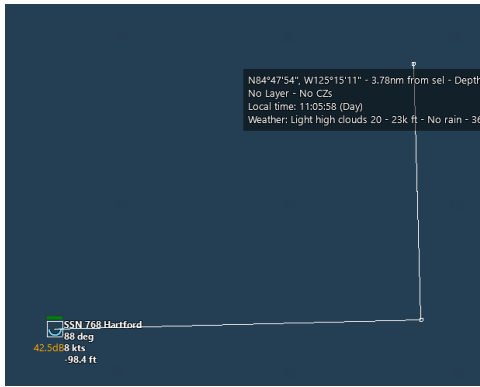
Loading the map, we can see:



The markers to the East of us represent the possible submarine patrol area and are where we need to start looking. We have to get there as quickly as possible so we will order our vessel to the patrol zone at full speed and maximum depth possible. Once we get to the patrol area, we'll prepare our sprint and drift patrol after listening. While it is tempting to start our patrol in the upper left or bottom left corner, bisecting the patrol zone will improve the chances of us finding our quarry.



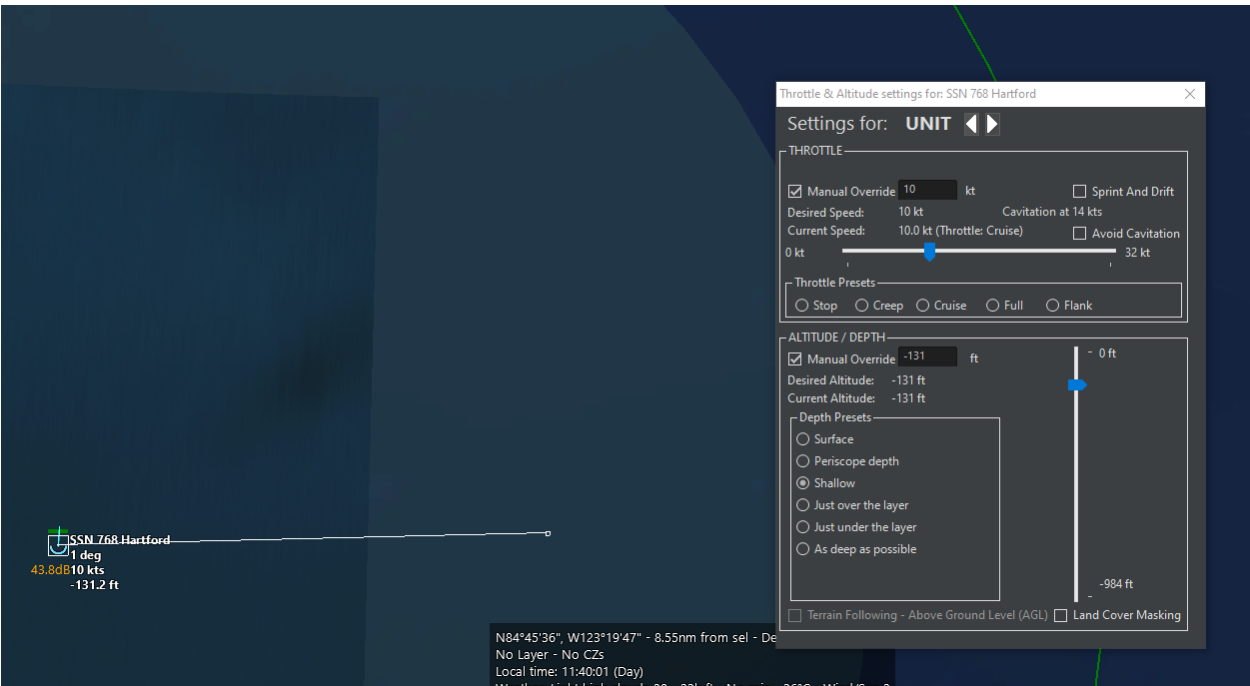
Our towed array is trailed behind and below us, but our own submarine noises will make it hard to hear 'through' us. Although we have a powerful sonar array on the fore of the submarine, we'll want to swing around often to listen.



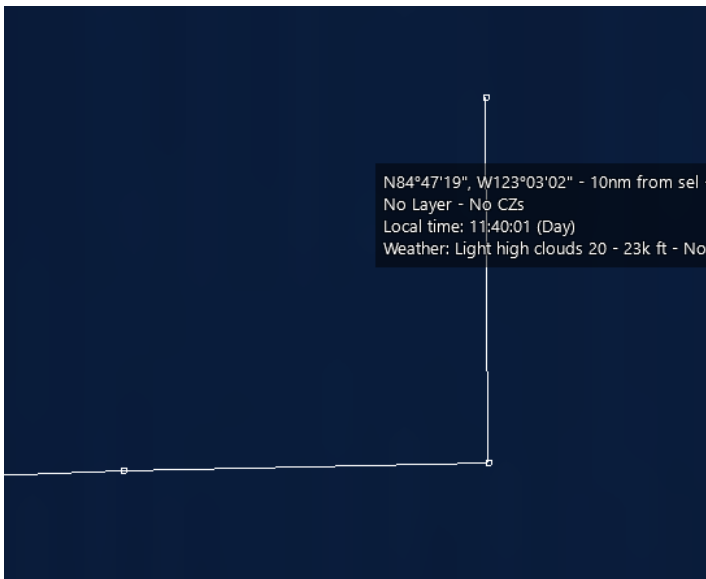
At anything under 10 knots, we have a good balance of speed and the ability to hear things around us. The faster we travel, the more noise we will make and the harder it will be to pick out the sounds of quiet submarines in our vicinity.

At this point we may encounter a possible contact, but this is unlikely given the size of the search area. If we do get the distinctive unknown subsurface contact, we can start working out our first shot immediately. Otherwise, it's time to set up the sprint and drift.

Our method will be to sprint about 10 nm at maximum depth and then drop our speed to 10 knots and then return to shallow depth for five minutes of listening and a quick 90 degree turn to clear our own baffles. 5 minutes at 10 knots works out to be about 0.42 nautical miles which means we'll spend a quarter mile going forward and then a quarter mile at the perpendicular course. Start by drawing a 10 nm course, then selecting our submarine and setting the depth to "As deep as possible" and the speed to flank.



Then click on the end waypoint and press F3 to continue the pattern forward about 0.25 nm and then north 0.25 nm



Now select the waypoint on the left (Waypoint #1 and set the depth to shallow and the speed to 10 knots)

Throttle & Altitude settings for: Navigation Waypoint ✕

Settings for: **WAYPOINT #1** ◀ ▶

WAYPOINT

Waypoint #1 for SSN 768 Hartford

DTG: 10 nm TTG: 58 min 53 sec

Fuel qty required: 0.0 kg

Description:

Type: Plotted Course / Navigation ▼

THROTTLE

☒ Manual Override 10 kt
 ☐ Sprint And Drift

Desired Speed: 10 kt

Current Speed: 10 kt (Throttle: Cruise)

☐ Avoid Cavitation

Throttle Presets

☐ Stop
 ☐ Creep
 ☐ Cruise
 ☐ Full
 ☐ Flank

ALTITUDE / DEPTH

☒ Manual Override Not Set ft

Desired Altitude: Not Set

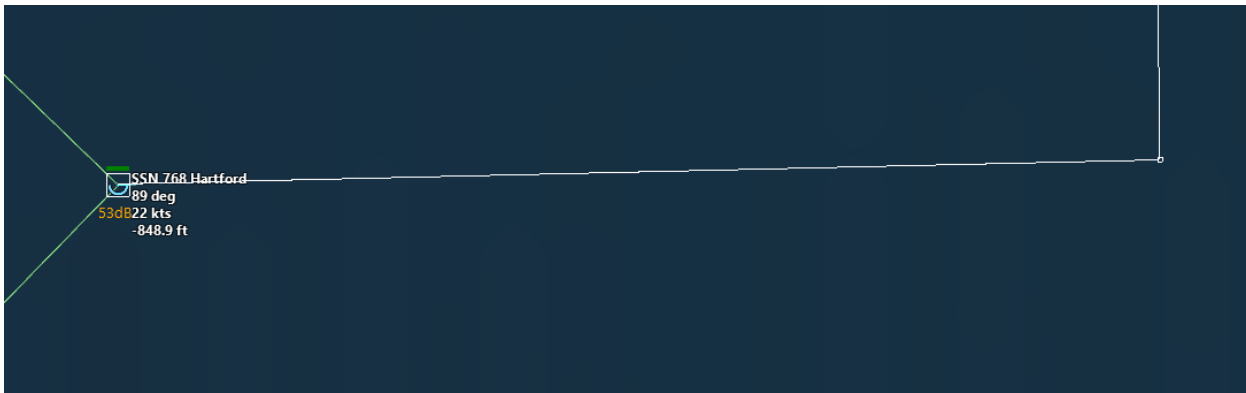
Current Altitude: -131 ft

Depth Presets

☐ Surface
 ☐ Periscope depth
 ☒ Shallow
 ☐ Just over the layer
 ☐ Just under the layer
 ☐ As deep as possible
 ☐ None

☐ Terrain Following - Above Ground Level (AGL)
 ☐ Land Cover Masking

Now our submarine will sprint to the waypoint, then come up to listen for a few minutes. All we have to do now is repeat the pattern on a macro level.

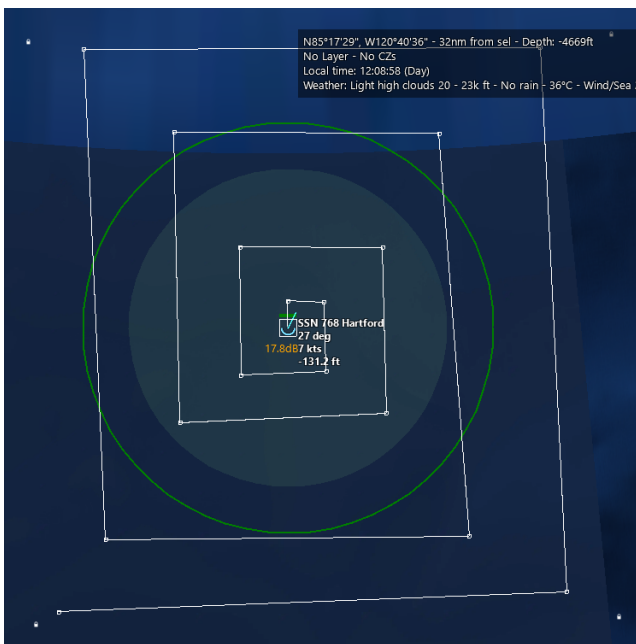


You can see as our submarine arrives, it heads back up to the surface for a listen. Do not be surprised if there is nothing there as sonar has limited range. We can continue this pattern of sprinting and listening at a much larger level. At this point, we have to consider what kind of search pattern we are going to use now that we have our depth and speeds worked out.

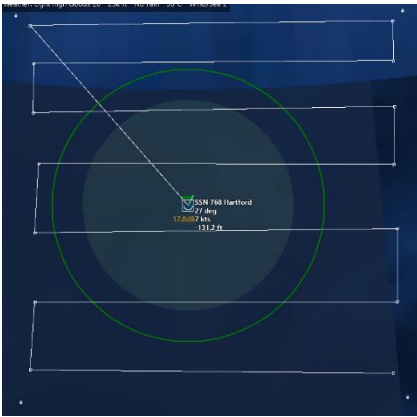
In ASW, there are several basic patterns we can use:

1. Expanding box: make progressively longer legs away from a point in the shape of a 'square spiral'
2. Ladder: Track back and forth up and down an area similar to the pattern someone mowing the lawn takes
3. Sector: Head to an area and execute several 120 degree turns before offsetting to another.
4. Zig-Zag: As in a box, but without the square corners

Expanding box:



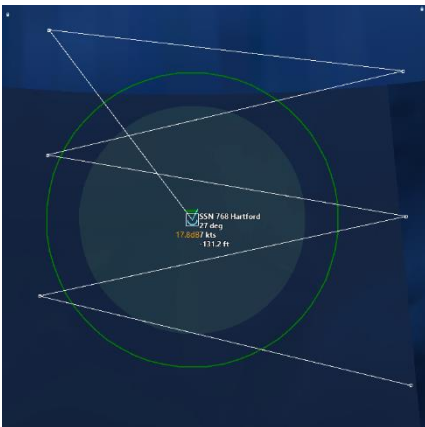
Ladder:



Sector:



Zigzag:



Notice how each pattern has a different level of coverage versus time spent. The ladder is very effective, but it takes the longest to execute while the sector search doesn't cover as much, but it allows for quicker transit. The zigzag gives us a good balance between transit time and coverage. The expanding box is very useful if we were dealing with a stationary target, but since we're dealing with moving targets, the time spent going back over the original terrain won't help us.

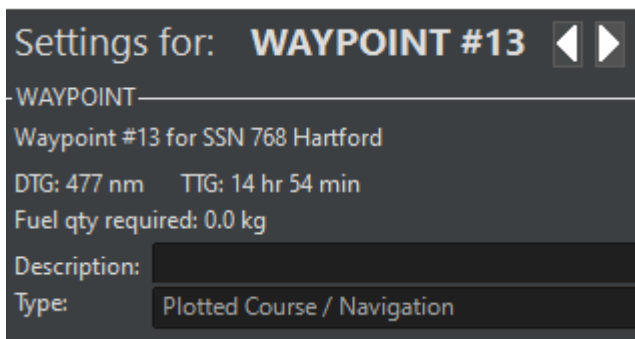
Unfortunately, what makes our decision tougher is our need to sprint and drift along the length of the pattern we create. We can either meticulously place each waypoint along our route, or we can opt for manual control of the submarine. Manual control also means we can react to anything we find along the route without having to redraw *our entire route*.

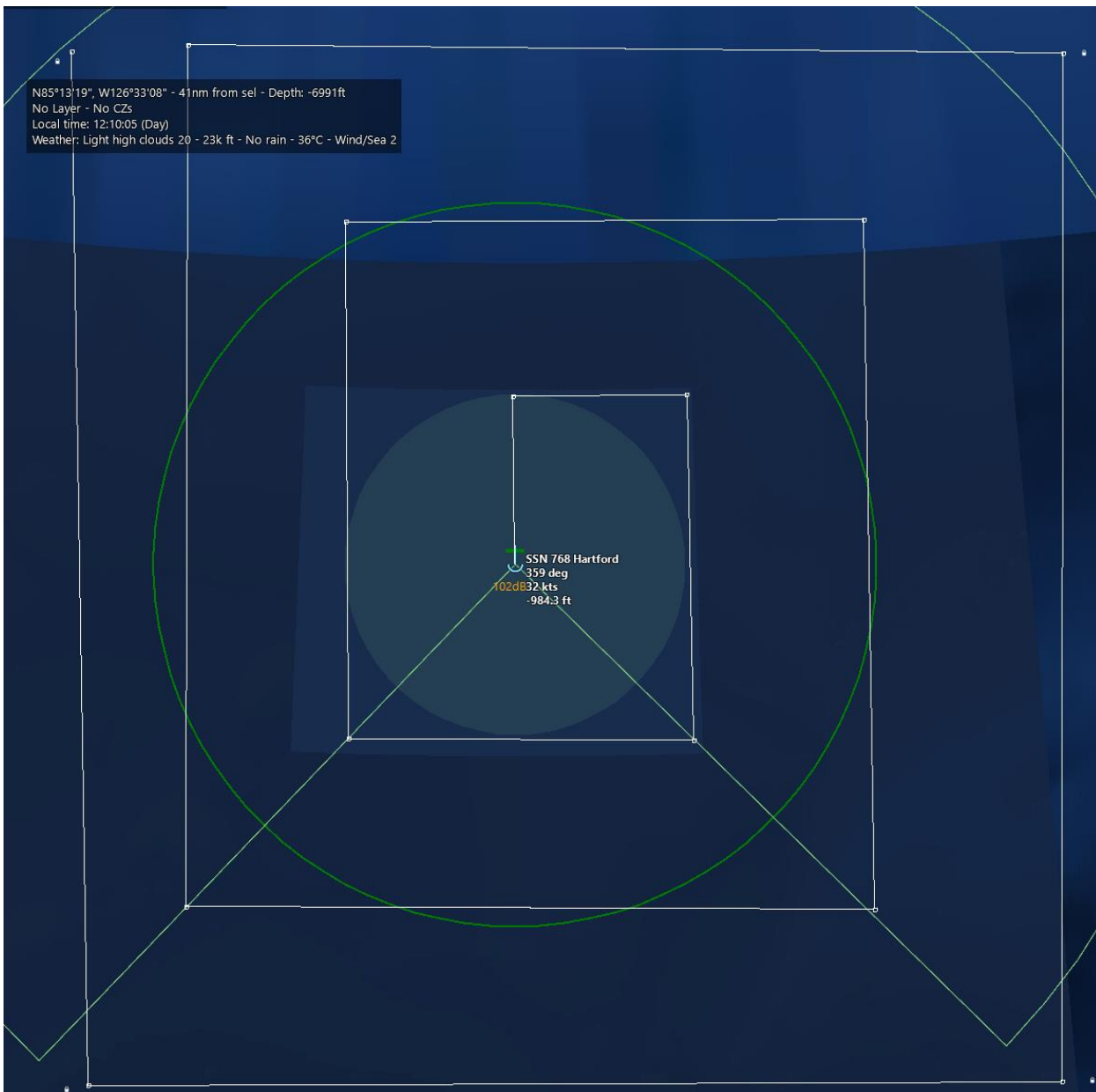
When trying this scenario, experiment with the different patterns. For the purposes of this tutorial, we'll opt for the expanding box and use manual control of our submarine. To make things a little easier for us, we'll build a simple system to keep track of when to sprint and when to surface and listen:

Time past the hour	Depth	Speed
+0:00	As deep as possible	Flank/32 knots
+0:20	Shallow	Creep/8 knots
+0:30	As deep as possible	Flank/32 knots
+0:50	Shallow	Creep/8 knots

This is also handy in case we encounter an enemy submarine and we need to determine where we were in the cycle.

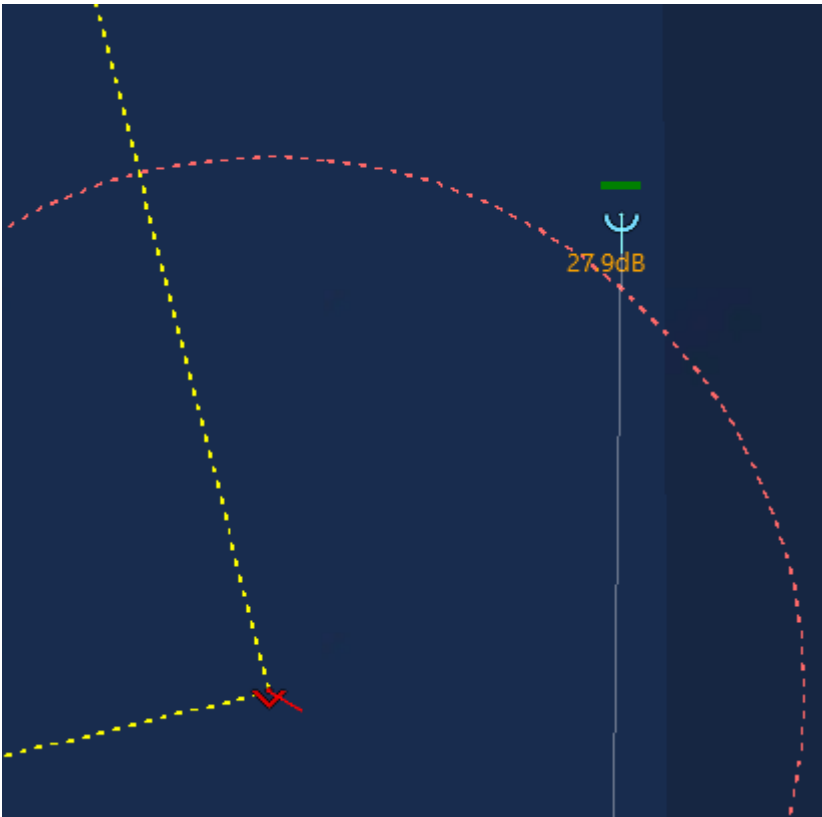
After setting up our pattern, we can see it will take us about 14 hours to cover the entire shape at flank, and with our adjustments we can expect that to be pretty close to 24 or so.





Our drawn pattern.

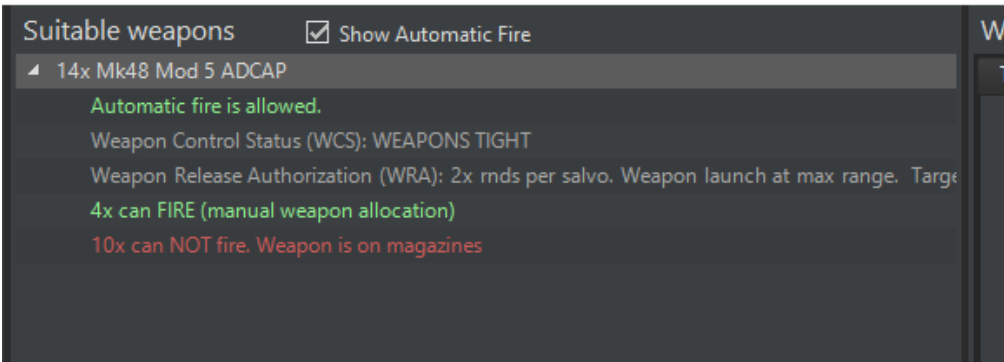
Now all that is left is to make our adjustments as we go along. You'll now see what ASW is also known as 'awfully slow warfare'. It may take us a literal two days in game time and still not see one of the four enemy submarines, or we will see all four at once, that's why it is so challenging.



ALARM! We've got something during one of our sweeps. Our sonar team was on their a-game and worked this contact out to be our quarry! This was incredibly lucky on our part, but just because we have a positive contact and ID, we still need to engage the enemy submarine.

Our range to the target is 3.37 nautical miles meaning our 50 knot torpedoes can cover that distance in about four minutes. The moment we fire, the typhoon will likely drop a rack of their own torpedoes down our bearing, but if you recall, their range is only 5 nautical miles giving us plenty of time to get outside of their detection range.

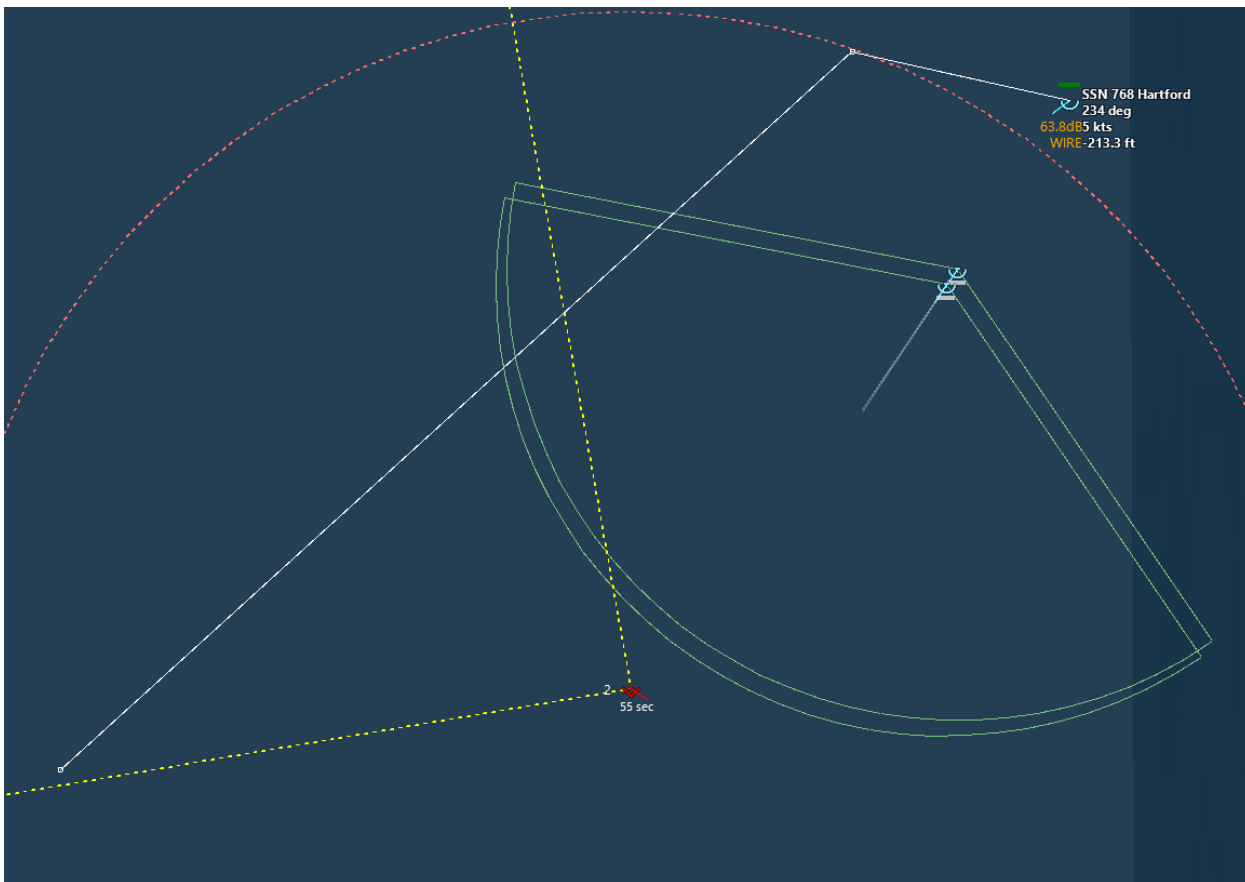
Hold down shift+F1 and left click on the target



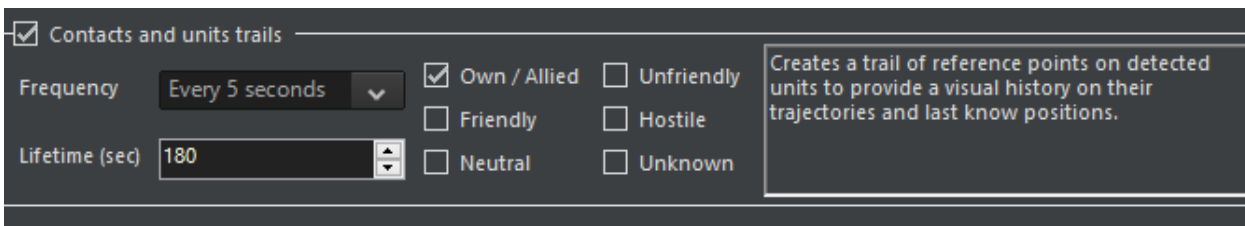
Although it is tempting to fire all four torpedoes at the target, typical submarine doctrine calls for always holding one torpedo in reserve for self defense. After all, there could be a Russian sub right behind us!

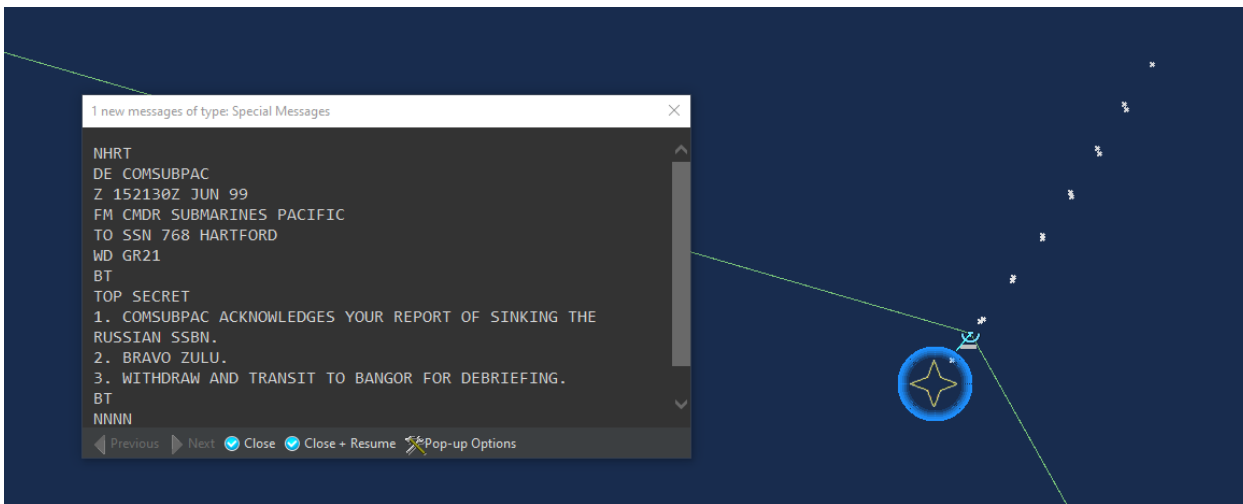
Since our range is so close, we can take a direct shot rather than firing a carefully aimed spread. Remember the default reaction to hearing a torpedo is turning and running from it meaning if we *miss*, we now have a target that is several miles away from us and out of range from a follow up shot, so we'll have to pursue them *remembering that we will be making a lot of noise in the process*. We'll have to run deep and fast and constantly track where our target is.

Fire two of the torpedoes, and notice immediately that you lose sight of the Typhoon. This is because your torpedoes are the loudest things in the water and they are between you and your target, you can either manually steer the torpedoes out of your line of sight (they are wire guided and can be controlled with the plot course command), or you can beam them until you reacquire the target. Remember they will run away from the incoming torpedoes, so we need to be where they are going.



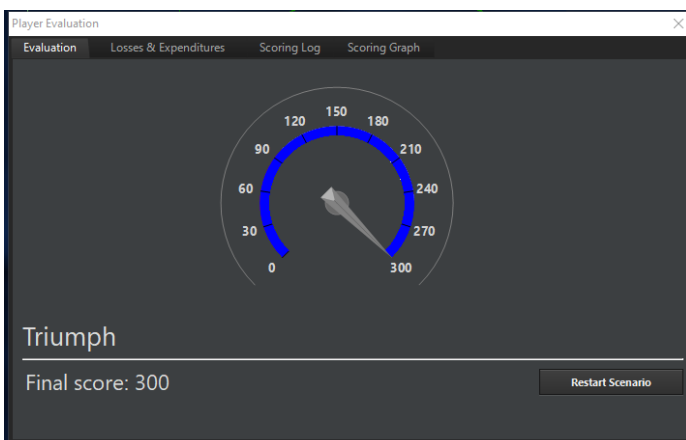
Since our torpedoes are so noisy, we can easily track the position of the enemy sub by the direction the torpedoes point and their path of travel. Turning on unit trails is a great way to visualize the path of the weapon. This is under options -> Map Display.





Those Mark 48 torpedoes are incredible weapons and you can see how it was a very clean kill leaving us with one torpedo still tracking. That torpedo can turn and *come back to us* if we are noisy, so be careful not to get too close to it. A common strategy here is using the wire to plot a course of the weapon away from our own submarine, but this can cause collateral damage!

In this scenario, it is very tempting to keep hunting for the other Russian submarines, but an order is an order. The BRAVO ZULU is a naval shorthand for 'well done'. Getting out of this area is quite easy since we can get down deep and run at creep speed until we are clear of the patrol zone. Remember that there are still several attack subs around us, and they may have heard us and are vectoring in to engage.



10.4.3 Conclusions and Discussion

Whether or not you were able to sink the boomer in the time allotted, the scenario provided you with a few great ways to approach anti-submarine warfare:

1. You can execute the mission perfectly and still miss your mark
2. If you can hit your opponent, they can hit you. This is not as true with aerial assets, but still relevant.
3. Detection is a function of speed and depth. The faster you are, the harder it is to hear but you will also be easier to be heard.
4. A search pattern is easier to execute when you have multiple assets at your disposal.
5. Sonar can be easily masked by your own vessel, weapons or other objects in the water making detection very challenging.

One thing we did not cover in the scenario write-up is the event that you are engaged by an enemy torpedo. The first thing you should do is launch one of your own torpedoes using a bearing only launch in the direction the torpedo was launched from while simultaneously marking where that weapon was when it was first detected. This will allow you to plan your re-attack after turning around and running at flank speed away from the torpedo. If a torpedo is fired at you beyond effective range, it is often easier to plot a course perpendicular from the incoming weapon to proceed out of the detection cone of it, also providing you with quality targeting information.

10.5 AAW Example – To be completed.

11.0 Appendices

11.1 Keyboard Commands:

This is a listing of various hotkey commands that can also be accessed in-game via the Help drop-down menu. For KEYPAD commands, NumLock must be off.

Basic

Spacebar, CTRL+ENTER: Pause/Resume Game.

CTRL+S: Save Scenario.

Tactical Map

CTRL+ [1...0]	Store quick-jump slot
[1...0]	Recall quick-jump slot.
Mouse Right-Click:	Center map on clicked point
Mouse Right-Click and drag:	Pan map
Up arrow, Num 8:	Pan map up
Right arrow, Num 6:	Pan map right
Left arrow, Num 4:	Pan map left
Down arrow, Num 2:	Pan map down
Z, Mouse Scroll:	Zoom in
X, Mouse Scroll:	Zoom out
V, PgUp, Num 9:	Switch unit/group view
(vert separator):	Select next unit
Backspace:	Select previous unit
T	Track selected unit
Del	Delete selected waypoint
CTRL+Ins	Add reference point
CTRL+R	Rename Reference Point
CTRL+Del	Delete Reference Point
CTRL+End	Deselect All Reference Points
CTRL+D	Range/Bearing Tool
End, Num 1	Toggle Illumination Vectors
Home, Num 7	Toggle Targeting Vectors
* (Star)	Toggle Datablocks
CTRL+M	Clear Message Log
CTRL+Shift+M	Message Log in Separate Window

Scenario Editor

CTRL+V	Toggle God's Eye View (see all)
INS	Add Unit
C	Copy Unit
Shift+C	Clone Unit
M	Move Unit
R	Rename Unit
Del	Delete Unit

Alt+S	Toggle Sides
CTRL+X	Map coordinates to clipboard
CTRL+F6	Add/Remove Aircraft
CTRL+F7	Add/Remove Boats

Function Keys

F1	Engage Targets (Auto)
Shift+F1	Engage Targets (Manual)
CTRL+F1	Bearing Only Launch
F2	Throttle+Altitude
F3	Plot Course
F4	Formation Editor
F5	Magazines
F6	Air Operations
F7	Boat Operations
F8	Mounts + Weapons
F9	Sensors
CTRL+F9	Unit/Group Doctrine
CTRL+Shift+F9	Side Doctrine
F10	Damage/Systems Status
F11	Mission Editor
CTRL+F11	Create New Mission

Own Units

U	Unassign Selected Units
G	Group Selected Units
D	Detach Selected Units From Group
A	Toggle Weapons Hold/Inherit for Selected Units
CTRL+A	Toggle Weapons Hold/Inherit for All Units
I	Toggle Ignore Plotted Course When Attacking For Selected Unit
CTRL+I	Toggle Ignore Plotted Course When Attacking For All Units
E	Drop Target
CTRL+E	Disengage (Drop All Targets)
L	Hold Position For Selected Units
CTRL+L	Hold Position For All Units
Shift+ [	Drop Passive Sonobuoy Above Layer
[	Drop Passive Sonobuoy Below Layer
Shift+]	Drop Active Sonobuoy Above Layer
]	Drop Active Sonobuoy Below Layer
Shift+D	Deploy Dipping Sonar
O	Display Order Of Battle+Contacts Window

Contacts

P, PgDn, Num 3	Drop Contacts
----------------	---------------

H	Mark Hostile
CTRL+H	Mark Unfriendly
N	Mark Neutral
F	Mark Friendly
R	Rename

Misc

CTRL+Shift+C	Lua Script Console
CTRL+C	Copy Lua values of selected unit to clipboard
CTRL+Z	Copy Lua values of highlighted RPs to clipboard

11.2 Custom Overlays 101

To add overlays to command you must first use third party software to generate geo-referenced files and then you can upload them to your command using an in-game interface.

For demonstration purpose we are using a modified GMAP as we've had good experiences with it. We do not guarantee its future functionality nor are we in any way responsible for the accuracy of the open source imagery.

Download the modified GMAP (<http://www.matrixgames.com/forums/tm.asp?m=3530108/>) and unzip to a location you'll be able to find.

To run the program, click the Demo.WindowForms.exe file.

When launched you can manipulate the map with your mouse and change the imagery type using type drop down. In general, it is best to find what you're looking for using maps and then go to the satellite imagery (Google satellite, Bing, etc.) to actually take the picture. I would even suggest going to Google Earth or finding the exact coordinates in Google first to save some time. You can also overlay any of these map types although the satellite imagery looks best.

Keep imagery small (no larger than a city) to avoid potentially serious stability and performance issues.

When you're ready to capture something to use as a file:

1. Press the Alt button and left click mouse drag the area you would like to capture. You'll know you're successful when it is highlighted with a blue box. When satisfied that you've captured what you want then press the get static button on the right.
2. When the static map maker dialog appears:
 - a. Set the Zoom level to 17
 - b. Check the Make Worldfile check box
 - c. Push the Generate button
3. When complete you'll notice that two files have been generated; one is a .png and the other is a .pgw.
4. You can further edit the files in any image editing program if you choose.
5. To import into COMMAND, launch the game, create your side and click the Custom Overlay button and when the Custom Layer Manager appears press the Add Layer button. This will launch a windows Add File dialog.

Please choose the .png file and allow the display time to load. Once loaded, you will see it in the bottom list. The most recent overlay will be highest on the list.
To remove an overlay, simply select it in the Custom Layer Manager and press the Remove Selected Button.

11.3 Database Edit Capability in the Scenario Editor

COMMAND offers several tools to allow users to edit facilities and naval units in the game. Users can add weapons mounts to a facility or naval unit by pressing the “Add Mount” button in the weapons dialog.

This launches the Add weapons dialog.

To add a weapon:

1. Search for the weapon you would like in the mount list at the bottom. To sort the list click on the column headers (ID or Name). You can also filter by a search term by checking off the filter by keyword check box and entering text in field above the mount list. Once you find the mount you would like to add select it.
2. Use the arcs checkbox diagram to select the weapon’s firing arc. If you’d like 360 coverage check them all off. Photographs of similar weapons give a good idea for how the arc is like. Surface-to-surface missile launchers tend to have narrow arcs, while forward-mounted gun turrets cannot engage in the rear of the ship.
3. Press the “Add Selected” button and now the weapon will appear in the weapons dialog.

To remove a weapon:

1. Select the weapon in the weapons dialog.
2. Press the remove mount button. The weapon should now be removed from your list.

Users can also change the weapons within an existing mount by removing existing weapons records and adding new weapons records. You can do this by:

- Selecting the weapon’s record you would like to remove by selecting it and pressing the “Remove Weapon Record” button.
- Pressing the “Add Weapon Record” button which will display the “Add Weapon Record” dialog.

To search for a record you can click on the table header or you can filter by keyword by checking off the Filter by keyword checkbox and typing in a text term in the provided text box.

To select a weapon’s record select it within the list in press the Add selected button.

Adding Sensors

Users can add a sensor to a naval unit or facility by pressing the “Add Sensor” button in the Sensors dialog.

NOTE: To add a new magazine to a unit you will need to add it with a configuration file. You can find instructions on that in sections 7.1.6-7.1.9 of the game manual.

To Add a Sensor:

1. Search for the sensor you would like in the mount list at the bottom. To sort the list click on the column headers (ID or Name). You can also filter by a search term by checking off the filter by keyword check box and entering text in field above the sensor list. Once you find the mount you would like to add, select it.
2. Use the ARCS checkbox diagram to select the sensors detection and tracking/illumination arcs. If you’d like 360 coverage check them all off.
3. Press the “Add Selected” button and now the weapon will appear in the sensors dialog.

To Remove a Sensor:

1. Select the weapon in the weapons dialog.
2. Press the “Remove Mount” button. The weapon should now be removed from your list.

And remember...

When making changes, the important thing to remember is that any “full” scenario rebuild will delete your edits. To retain them you’ll need to add changes to a scenario configuration file. You can do this by generating a so-called “delta file” which shows the changes from the “stock” configurations, and then applying them to the configuration file. Specific instructions on how this works can be found in sections 7.1.6- 7.1.9. Keep in mind that our DB editors always reserve the right to change record numbers so it is best to keep up with changes.

11.4 Common Air Units

Here is a list of common air units found throughout COMMAND.

10.4.1 Early/Mid Cold War

These are “first” (Meteor, F-80), through “Third” (F-4, MiG-23) generation aircraft. These are the mainstays of scenarios using the CWDB.

- Meteor. First-gen British fighter
- F-80. First-gen American fighter
- MiG-15. Early Soviet fighter, the bane of the USAF over Korea.
- F-4. American fighter. A multirole aircraft that ended the frantic “Moore’s Law For Jets” period of 1945-1960. Very versatile.
- MiG-21. Soviet point defense interceptor with limited ground attack capability.
- MiG-23. Soviet swing-wing fighter capable of launching radar-guided missiles. The MiG-23BN is a ground attack version.
- F-5. Lightweight, no-frills western fighter.
- Su-17/22. Soviet high-speed ground attack aircraft.
- F-105. “Thud”-basic Vietnam-era ground attack aircraft.
- F-104. High speed point defense interceptor, can be considered a western analogue to the MiG-21.

10.4.2 Late Cold War/Modern

These are “fourth-gen” aircraft of the late Cold War to modern period, in use widely around the world. While they may look similar, many of them are designed for quite different roles.

- F-15A/C. Purpose-built air superiority fighter, “not a pound for air to ground.”
- Su-27. Similar to the F-15A, but has a small ground attack capability.
- Su-30/35. Upgraded versions of the Su-27. The Su-35 is the most advanced version.
- F-15E. Although the same shape as the F-15A/C, it’s a highly capable strike fighter with much air to ground capacity.
- F-16. Lightweight multirole fighter. Intended as a “jack of all trades” that could be built and fielded en masse. Goes from short-ranged Sidewinders and dumb bombs to radar-guided AMRAAMs and

precision weapons assisted by advanced targeting pods, but keeps its role as the “low” in the “high-low” mix.

- F/A-18. Roughly equivalent to the F-16, only for carrier wings. Has an earlier ability to launch radar guided missiles.
- MiG-29: A point defence interceptor, more of an heir to the MiG-21 than the multirole Western fighters it externally resembles. Built to operate inside an integrated air defence system. Has a small air to ground capability.
- F-14. Carrier interceptor with later air-to-ground capacity installed.
- Tornado IDS/GR: Long-range ground attack aircraft.
- Tornado F/ADV: Bomber interceptor, not intended to engage other fighters.
- A-10: American ground attack aircraft geared towards close range support.
- Su-25: Soviet ground attack aircraft also geared towards close-range support.
- Su-24: Long-range ground attack aircraft.
- E-3: AWACS.
- A-50: AWACS similar to the E-3.
- Harrier. V/STOL Aircraft. Low-performance but highly deployable. Frequently deployed from smaller aircraft carriers and amphibious ships as a “one eyed man in the land of the blind”.
- Yak-38: Soviet counterpart to the Harrier. Even lower performance.
- Rafale: Multirole “4.5” gen fighter. The Rafale M variant can operate from carriers.
- Typhoon: Another “4.5” gen fighter, more optimized for air to air than the Rafale but still has multirole capability among some users.
- Gripen: Swedish light multirole fighter with excellent rough-field capability.

10.4.3 Aircraft Organization

The average size of a fighter squadron ranges from 12 aircraft to 24 (a USAF squadron at full paper strength). Multiple squadrons form a “wing” (western) or “regiment” (Soviet), while multiple wings/regiments form a group/air division.

11.5 Common Sea Units

In the same vein as the air units list above, a list of common modern naval units.

- Nimitz: American supercarrier.
- Kirov: Soviet/Russian missile battlecruiser, intended as a fleet flagship and heavy missile launcher.
- Kuznetsov: Soviet/Russian aircraft carrier. More like a Kirov with an air wing instead of S-300s than an American supercarrier.
- Arleigh Burke: American multirole destroyer.
- Perry. American low-end anti-submarine frigate.
- Ticonderoga: American anti-air cruiser.
- Slava: Soviet/Russian missile cruiser.
- Udaloy: Soviet/Russian anti-submarine destroyer.
- Spruance: American high-end anti-submarine destroyer.

11.6 Common Naval Tactics

The important difference between the Cold War foes in terms of their naval systems is that the American/NATO systems trend towards the *tactical* defense but the *strategic* offense, while the Soviet/Russian ones are, if anything the opposite.

Thus an American cruiser or destroyer will generally (not always) have an anti-surface armament of only light Harpoons and its SAMs pressed into a horizon-limited anti-shipping role. Its main job is to protect the carrier/amphib. That high value unit, on the other hand, has the role of projecting power around the world. This can mean anti-shipping power in the form of its air wing attacking an enemy fleet with long-range missiles, or anti-ground power in the same air wing bombing enemy units or covering an amphibious landing.

A Soviet ship of a similar period and displacement will tend to have a much heavier anti-surface armament and (generally) a weaker anti-aircraft one. This is to make up for the lack of carriers. Soviet naval doctrine was highly defensive-the keystone was simply defending their nuclear missile submarines long enough for them to launch.

Even large high-value units were focused towards this goal-the Kiev and even Kuznetsov/Ulyanovsk carriers are built as large antisubmarine carriers. To a degree, a large portion of the “offensive” slack was taken up by fleets of large land-based aircraft in the Soviet Naval Aviation such as Tu-16s and Tu-22s. These carry monstrously large ASMs and have long range.

Weaker factions than the Cold War superpowers are constrained more heavily by their equipment. One need only look at the Falklands, where the British Invincibles (which grew out of a helicopter carrier, carried Harriers initially just to shoo away patrol aircraft, and lacked AEW at the time) faced Argentine air forces (who had a severe shortage of both air-launched Exocets and tankers that had a massive effect on their operations). A regional power with a large PGM-capable fleet can do things that a country with a handful of light ships and old aircraft cannot.

Regardless of what the strikers consist of, naval warfare is based on massing as much of them at the target as possible. It is and always has been a very deterministic type of war.

It can be boiled to three simple maxims:

- See, and try not to be seen. Whoever finds the target first will usually strike it first.
- Whoever strikes the target first will almost always win.
- Half measures will almost always result in nothing being gained. Concentration of force is necessary.

11.7 Comms Disruption & Cyber Attacks

Most wargames and simulations accustom their users to the idea of omnipresent, instant, always-on communications. The battle may be lost – your forces & assets may be in disarray – but you are always in control of them and aware of their whereabouts, activities and condition. You are always in the know and in power. The pawns in the chessboard are always at your disposal.

But what happens when the lights go out? When the only sound on the radio is static? When the datalink is dead?

This is fast becoming an ever-important question, both because the western armed forces are increasing their reliance on distributed, hyper-connected force concepts (the F-35 and the US Navy's NIFC-CA being fine examples) and because potential peer competitors, fully aware of this trend, are rapidly improving their electronic attack and cyber-warfare capabilities. The ever-evolving chess game of electronic warfare has never been limited to the radar bands, but more recently communications and datalinks are becoming increasingly appealing targets.

Command handles this aspect of operations explicitly. We want to show players both the “how” and the “why” of communication lines breaking, and the practical effects of units being isolated from their parent side's common picture. A unit that is out of comms will have a small set of text reading NOCOMM next to it to indicate its situation.



Causes...

In order to enable this feature, the scenario author must ensure that under “Scenario Features and Settings”, the “Communications Disruption” box is checked.

A unit can have its communications disrupted in a number of ways:

If all its onboard comm devices and datalinks become non-operational (damaged or destroyed):

This frigate is about to become very lonely

Through the Lua scripting API: As an example: `ScenEdit_SetUnit({Name="USS Vigilant", OutOfComms="True"})`. This is a very versatile technique as it can be used to model any number of factors, both man-made and natural, that can cause a unit to go “off-grid”:

A submarine hooking up to HF or satellite communications when it goes to periscope depth, and breaking contact again when it submerges.

A satellite linking only temporarily with its ground station to dump its intelligence product and receive new tasking instructions, then again going in isolation in space (*contrary to popular fiction, even modern intelligence satellites are rarely, if ever, in constant link with their ground control stations*).

A cyber-attack directed at the internal comms infrastructure of the targeted platform. (*Your comm devices may be physically untouched, but the server at the core of the comms exchange just got taken over. Tough world!*)

Physical incapacitation (damage/destruction) of a critical C3I node leading to other units dependent on it going offline (*remember that many of the first-night targets in Desert Storm were headquarters, C3 bunkers and comm buildings? You can recreate why this was a critical action*).

Through the scenario editor GUI, by selecting the unit and marking it as “out of comms”.

This is less powerful than using the Lua API but a simpler and faster way for quick setups. So for example you may use this to configure a unit to be “off grid” at the start of a scenario, and follow-up with a Lua script depending on some later action that changes this unit’s connectivity.

By jamming its comm devices, using communications jamming equipment similar to existing OECM systems. [NOTE: Integrated comms-jamming is a feature reserved for the *Professional Edition*. However, with a bit of creative work you can use the Lua API to approximate this in the commercial version of Command.]

...and Effects

So what are the results of a unit going “off-grid”?

The most obvious effect is that the unit is no longer under your positive control. You no longer know where it is currently located (you only know where it was when it last checked-in, and if any of your other still-connected assets manages to make contact with it, this “last datum” is updated). You don’t know what it is doing, what its fuel, weapons or damage status is. You don’t know if it is peacefully loitering with not a care in the world or if it is fighting for its life. You will know its fate with certainty only when it comes back to its base – or is destroyed first.

“Last we heard from Garry he was somewhere there... and that’s all we know.”

For its part, the cut-off unit loses all the benefits of the common side-wide operational picture; its situational awareness now reaches out just to the limit of its own sensors and no further. It has no idea what is going on “out there”. It can still detect, investigate and prosecute contacts on its own and proceed with its assigned mission if it has one, but all the benefits of mutual support are gone. Cooperatively patrolling a large area to split up areas? Nope. Efficient fire coordination? (“You shoot bandit #1 and I’ll take #2”) Forget about it. Even worse, it now has to be really careful with anything that shows up on the scope. Is that new contact a friendly or an enemy? You’d better hope its RoE and doctrine settings take such a situation into account – or prepare for blue-on-blues!

Meanwhile, Garry is now alone and trying to adjust to the sudden loss of SA from the nearby AWACS.

Notice the only contact still “fresh” is the one that the F-15 is actually detecting on its own; the other contacts, hitherto provided by the E-3, are now deteriorating fast.

Units that lose their comms connectivity retain local copies of the contacts that were available to them before they went offline; essentially they inherit a snapshot of their parent side’s theater picture at the moment of their breakaway. However, without the benefit of information exchange with their parent network, this snapshot immediately starts to lose its currency and most of the contacts will soon vanish unless refreshed by the unit’s own sensors. *(It’s like walking down a busy street, taking a last look around and closing your eyes. The longer you remain blind, the less relevant & useful your last memory will become.)*

Units that manage to get through their “isolation” and re-join their side comms network share their contact information. You can use this to model things like film-return satellites *(the Russians still use them!)*, a submarine sharing its intelligence take after rejoining its battlegroup etc. If the parent side already has these contacts, the updated information *(including BDA – very handy!)* is merged and used to refine the contact information.

As mentioned, the player has no control over his “disconnected” units. However, playing out a scenario in ScenEdit mode (which is essentially “cheating” but also very useful for analysis) affords an additional ability: To literally jump into the cockpit/CIC of the isolated unit and experience its “loneliness” first-hand (**Editor → Isolated POV view**). This is an excellent way to understand how a comms-isolated unit perceives its environment and reacts to it (which is usually quite different from when it operates as part of an integrated network). Quickly switching between “side-wide common picture” and “isolated POV view” can be a real eye-opener as to the value of a well-connected battleforce and the hazards and challenges of “comms off”. It is simple enough to say “we’ll not transmit so the bad guys cannot sniff us out”. But how well can you really fight in the dark?

Living with vulnerable comms

Players using this feature will probably have to adjust to a different mindset as well as absorb the new options it affords them. A few random thoughts:

- Non-kinetic options are now significantly more expanded. Forget jamming radars; now you can do some real sneaky tricks. It is remarkable how much more brittle and inefficient, say, an IADS becomes when you can selectively take any of its critical nodes offline. *(Michael Scofield would love to be able to brick the guards’ phones during his break-out/break-in escapades)*. Put another way: Going ultra-stealth or swarming cruise missiles over every radar & SAM site are no longer your only, or even your best options.
- An initial tempting thought was to freely use comms disruption as a standard built-in feature to all existing scenarios. It was soon discovered, in hilarious ways, how this can wreck scenarios that were not designed with this factor in mind. As an example, in the standalone scenario “Duelists”,

we “flipped the switch” on the Soviet surface group just to see what would happen. Because the group had a standing “forbidden zone” around it that automatically marked violators as hostile, and in combination with very liberal “shoot first” ROEs, the ships in the group almost immediately proceeded to blow each other out of the water with righteous fury, with even a distant Oscar-class sub joining the action with its beastly antiship missiles. It really was blue-on-blue in its most pure, raw form.

- Apart from the obvious step of making this new feature optional so that existing scenarios can function as before, we also added a lot of “reasonable smarts” to the AI crews so that events like the above Soviet shootout were less likely to happen. So for example, when detecting a new contact close to where a friendly off-grid unit was last known/reported, the side-wide AI is now a bit more careful even if the new contact is a violator of a forbidden zone (“easy on the trigger, guys... maybe it’s Garry after all”). Likewise, cut-off units compare their new contacts with the most recent datum of their known comrades and will ignore contacts that seem to be right on top of where their buddies were when they last heard from them. After putting together such refinements and re-trying the Duelists test, the isolated Soviet units were now significantly more restrained on how they prosecuted sudden new contacts after their breakaway.

(On the flip side, if you are the one creating the chaos, this creates excellent “false flag” opportunities for exploiting it. For example, if you can swiftly and unobserved dispatch comms-isolated units that are part of an enemy distributed force, the longer you can remain in their general location the longer it may take for their friends to get suspicious of you. Cloak and dagger fans rejoice!)

- Comms vulnerability really drives home the dirty little secret of most network-based CONOPS. People talk about swarms, distributed lethality and all the rest of current jargon – but all of this assumes reliable communications. This also reinforces the importance of sufficient individual capability: A modern warship, even when comms-isolated, is still a powerful unit. It has a fair chance of accomplishing its mission even without the obvious benefits of cooperation with its consorts. On the other hand, a “swarm” unit (be it a small speedboat, a drone etc.) that relies on the “my strength is in the pack” principle? Take comms off and its power drops precipitously (it is not an accident that comms-jammers are rather more favored for counter-drone defence than hard-kill measures).

Some other scenario-editing ideas for taking advantage of this feature:

- Special actions, with units on the opposing side getting their comms knocked out. This is used to simulate a computer network attack, but could also represent saboteurs physically disrupting the comms network.
- A “unit enters area” trigger where a friendly unit representing a jammer (ie, an EA-6B Prowler or similar) arrives on station. The script is fired, and representing its jamming capabilities, units on the opposing side have their comms taken out.

- A “unit enters area” trigger where run-off the mill friendly units get their comms knocked out, forcing them to continuing prearranged missions or move aimlessly in circles until they run low on fuel and return to base. This, especially in earlier-period scenarios, simulates units outrunning their lines of communication. It can also be used to simulate submarine operations.

11.8 Tacview

To enable the use of Tacview in Command if owned (NOTE: TACVIEW MUST BE PURCHASED SEPARATELY, AND THE ADVANCED OR ENTERPRISE EDITION IS NECESSARY):

- Download and install Tacview at <https://www.tacview.net/>
- Go to “Tacview” under Game Options, locate and select the Tacview executable (the default installation folder should be C:\ProgramFiles(x86)\Tacview)
- Go to the “View” drop-down menu and select 3D View (see 6.2, View)

To center the 3D Tacview map on a certain unit, either left click on them in the main Command window or Control+Left Click on them in the 3D View window. They can be zoomed in and out or rotated via mouse grip.

To add extra models: First, construct a suitable 3D mesh and place it in the proper folder (see here for instructions: <https://www.tacview.net/documentation/3dobjects/en/>). Afterwards, edit the Excel file in [COMMAND INSTALLATION FOLDER]\Resources\Tacview\Associations_DB3000.xls and add a new row in the relevant sheet with at least the DBID and mesh file name of the object. Command will then read the entry and provide the filename to Tacview to render the object. This is currently possible only for DB3000 units.

With each update, more and more “official” Tacview models have been added.

To add more detailed elevation data & textures, see instructions here: <https://www.tacview.net/documentation/terrain/en/> .

NOTE: In early releases, Tacview displayed all units of all sides, even if the player’s current side could not see them, similar to the God’s Eye View mode. As of CMO 1.01, it has been changed to no longer be the case.

11.9 Detailed Design Notes

Note: This section shows design notes from <https://command.matrixgames.com/>. These are the raw comments from the Warfaresims team at the time of release of each feature. That means, as Command develops through the years, the comments can become outdated or even vestigial.

Overview of War Planner (nicknamed Tiny)

Operations Planner: Ever wished you had an ATO-like overview of all missions and operations planned or currently executing, their status and hierarchical priorities and dependencies? With units or even entire task

forces automatically switching from one mission to the next as objectives are achieved? Wish no more. The brand-new Operations Planner makes this, and much more, a reality.

Multi-Domain Strike Planner: Throw away your planning spreadsheets! You asked/begged/hostaged family members for it, and now it's here. Coordinate massive, complex strike missions with time-on-target, complex flight plans (incl. in-flight refueling) , multiple attack patterns and multi-domain strike combinations. "Bringing everything together on a strike is just too complex/difficult" is officially over as an excuse. If you don't master this, your adversary most definitely will.

Cargo 2.0: The sub-header for this feature is often "The Logistician's Nirvana" – and that should be your first hint. Transfer both combat units and also weapons, stores, fuel and any arbitrary material. Place your cargo on a multitude of different container types, from standard ISO-blocks to specialized boxes, each with its own peculiarities. Transload cargo at airbases, ports, railyards etc. in order to haul it over even transcontinental distances. Automate all this through cargo and (NEW!) transfer missions. Set up complex logistical chains from mainland factories all the way to the front line. Conquerors from Napoleon to Alexander would have given their right arm for such a tool – and you get it for free. Who ever said life is fair?

Area & Reference Point Manager: Areas, zones, ref-points and overlapping fields oh my! A centralized way to manage anything and everything related to reference points.

Custom Environment Zones: Multiple & moving weather fronts? Check. Bend the laws of physics on a localized area? Can do. Specify carefully hand-picked weather, terrain and other environmental properties in order to test or compare sensors and other environment-dependent components? Yup. Unleash your inner nature wizard with this puppy.

Air combat mechanics overhaul: Boost-coast AAW missiles (ie. the vast majority of them) now correctly accelerate to their maximum speed by their boosters and subsequently coast over the rest of their flight, trading altitude for speed (and vice versa) while also shedding speed due to drag – especially on sharp maneuvers. This makes them much easier to avoid at the edge of their envelope, where their energy reserves are depleted. Missiles with specialized long-burn motors have a decisive terminal-energy advantage over plain boost-coast systems. Virtual pilots are aware of these new dynamics and will exploit them to "drag out" incoming missiles, with beaming/notching as a last resort. To counter these counters, new additional WRA firing-range settings (including "No-Escape Zone") are available. A2A and S2A missile engagements are, as a result, both more dynamic and far more realistic now.

"Double-flame" time acceleration: "I wish my simulation runs had executed more slowly, I had time to spare" – said no-one on their deathbed. Aside from an array of general sim-speed improvements, this update brings a brand-new exclusive (and optional) time accel mode: "Double Flame". This cranks up the virtual timeslice to 5 seconds, massively speeding up simulation execution. "Hold on", we hear you say, "bad things start to happen when you get so coarse in your timeslice". You bet they do. So how did we solve it? Find out in one of our follow-ups.

HGVs and D-EMPs now available: Yes Dorothy, the previously available only in CPE hypersonic glide vehicles and directional-EMP weapon types are now available in the commercial version too. Yes, they can be pretty useful if used correctly – especially in coordination with other, more abundant assets. No, they won't save you from certain doom if the rest of your ops suck. Treat them as magic saviors at your peril.

Palletized Weapons & Stores: Yes, we know you've all seen AFRL's videos on the Rapid Dragon concept. Yes, we know you drooled over the new possibilities. Newsflash: So did we. So now you can do that too. Is it awesome? You bet it is.

Passive Coherent Location System (PCLS) sensor, aka "Passive Radar": How do you detect and track stealth aircraft? One of the possible ways is to break out of the classic monostatic radar paradigm and embrace alternative solutions like PCLS. Are they omniscient? No. Do they have operational drawbacks, some of them quite severe? Yes. In the right conditions, can they detect stealthy aircraft that conventional radars are hopeless against? Yes, yes and yes. Find out how.

Intermittent Emissions: Radars and other active emitters no longer have to strictly choose between active and silent: You can now blink, and schedule how to. No scripting necessary! (But scripting still a very powerful option). Find out how, in our follow-up article.

Revised Mission Editor: "It's insanely better than before!" is the least positive comment we've heard about the revised ME layout. Now you too can experience what every beta player considers the finest visual experience since "Love Actually: The Directors Cut".

Revised Message Log: You asked (intensely and consistently...) for a fully filterable message log, dynamically enabling/disabling messages per type, while also retaining the "responsive nature" (show location, optionally show balloon etc.) of the messages. Oh and also keeping all the existing goodies (docked or free-floating, "raw"/waterfall mode, per-type coloring, balloons etc.). Our first reaction was "So, you essentially want Visual Studio's errors & warnings list?" Our second reaction was "Maybe this is feasible after all". Our third reaction was to go ahead and build it. And there was much rejoicing throughout the land.

Distinct Ground Units: As much as we keep telling you Command is not SPMBT or Combat Mission, you keep asking for distinct ground vehicles. So there you go. Command now offers individual ground units, in addition to the previous method of modeling them as aimpoint-facilities. You'll find ground units modeled with the same level of detail users expect of ships and aircraft: propulsion, fuel, mounts, sensors, etc. are all simulated, as are characteristics like armor (incl. ERA, slat armor, etc.). Ground units bring brand-new capabilities to the fight, such as amphibious vehicles capable of transitioning between land and water (with distinct speeds and fuel consumption in each domain).

Save/Load Doctrine & ROE Settings: Sorry, Cimmerian; crushing your enemies and seeing them driven before you is only the _second_ best thing in life. The first-best, is to carefully construct your Doctrine & Rules of Engagement settings for a specific side, mission, group or individual unit, and then export them to a text file where they can be freely re-imported for any other simulation object as well as edited, either by hand or in an automated fashion. With power such as this in one's hands, no prayers to Crom are necessary.

Overhauled Reaction Times: The differences in reaction times, and their effects, are now more critical than ever. All units use common-reference "Combat System Generation" ("Cockpit Generation" for aircraft) to model the modernity of their combat systems, combined with an "Ergonomics" value to handle intra-generation differences (the atrocious switchology on early missile-age aircraft will most definitely get you killed now). Older, WW2-era ships may take up to 5 minutes to engage a target, while Aegis cruisers fire in <20 seconds. Cold War fighters will be beaten to the draw by modern, fifth-generation fighters. Overmatch, that ever-elusive dream, is now possible – but beware, it goes both ways.

IR & Visual Sensor Improvements:IRSTs and high-mag cameras are no longer near-magical counter-VLO sensors. They may still be your best bet for detection, but you won't be volume-scanning for stealth fighters at >100nm anymore. (You can still spot/track them pretty far enough IF something/someone else first cues you there)

Radar & IR Stealth Improvements: Sensor improvements come coupled with a massive overhaul of signature modifiers in the DB, which significantly improve the realism of our stealth model by drawing clearer distinctions between shaping and RAM generations. We also added special DB "flags" to indicate the presence (or lack) of certain stealthy design features such as S-shaped intakes, exposed fan blockers, active cancellation, and stealth pylons. The overhaul also extended to IR modifiers, which now not only model whole-aircraft IRSS (distributed vs. conventional fuel tanks, low-E coatings) but also specific IRSS features such as shielded "anti-Strela" exhausts, masked exhausts, heavily masked / slit-shaped exhausts, and peak temperature reduction or "cool-air mixing".

Formation Presets: Tailor your formation's layout with a number of different presets (wedge, circle, echelon, line, diamond etc.) or make and store your own custom template. Customize facing and distancing to suit your preferences. Transitioning from parade to full-attack and back (perhaps in the middle of your parade?) has never been easier.

Bathymetry Layer: Visualize the terrain contours of the naval domain as clearly and as richly as you already do for the overland globe surface. A must for every facet of underwater ops, from sub/ASW hunts to mine warfare to UUV control to tracking whale migrations. Jacques Cousteau would have rightly wept.

Graphical Satellite Pass Display: The "Satellite Pass Prediction" window now offers a graphical representation of future satellite passes, sparing users the need to compare timeblocks to figure out which satellites will be overhead when. Who needs data grids when you can visually compare availability windows.... right?

Benchmark Mode: Looking to compare rigs? Command now includes a benchmark mode, which will repeatedly run any scenario you choose in "headless" mode and output performance metrics. Now your machine's sorry state will be plain for all the world to see.

Miscellaneous UI Improvements (Weapon Quick-Allocate, Slug Trails, "Barks," etc.): Quickly allocate weapons manually, without needing to bring up the (intimidating to some, apparently...) manual allocation window. Have units or contacts bark to each other (yes, you read that right). Make unit & contact movement paths & histories more obvious (and easier to visualize in a screenshot) by enabling "slug trails". Plus a few more tricks we'll see in detail on the UI-dedicated UI follow-up.

Weather effects on ship seakeeping: Sea state limits are no longer treated as hard lines. Smaller ships are now affected by progressively higher sea states, which will slow their max speed but not necessarily immobilize them. Moving from the QE to a small sloop now most definitely feels like it.

Aircraft Max Endurance: If you've ever used the "keep aircraft in the air indefinitely through repeated air-refuelings" cheat, first: SHAME ON YOU! Second: You won't be able to do it anymore. Aircraft are now limited in persistence both by their onboard consumables and also crew fatigue. And you wonder why everyone loves drones.

The Operations Planner <https://command.matrixgames.com/?p=5486>

Ever wished you had an ATO-like overview of all missions and operations planned or currently executing, their status and hierarchical priorities and dependencies? With units or even entire task forces automatically switching from one mission to the next as objectives are achieved? Wish no more. The brand-new Operations Planner makes this, and much more, a reality.

To fully comprehend this function, it is essential to have a good knowledge of the mission editor in general, and cargo missions in particular. Previously, in Command, a given unit could only be assigned to a single mission. If you wanted to assign the unit to another mission, you would have to manually unassign it from the current mission and then assign it to a new one.

Before we begin: Some nomenclature
Triggers

A trigger in the operations planner is a condition that gets checked each simulated second. If the conditions are met the trigger will execute a specific action. There are two possible actions: start a mission or tag a mission as satisfied.

Mission status

A mission's status in Command can be either Active or Inactive.

Active means the mission is evaluated by the Command simulation, but it doesn't necessarily mean that the mission serves a purpose or has units assigned to execute it.

Inactive means that the mission is completely ignored by the simulation until it becomes Active.

IMPORTANT: It is strongly recommended to leave all missions active, especially when working with the operations planner. Set mission as inactive only for draft missions or manually controlled missions.

Mission phases

A mission phase is a new concept introduced with the operations planner and is not related to the existing mission status:

“On Hold”:

The mission has not yet started, the units assigned to the mission won't execute the mission.

“Satisfied”:

The mission is considered to have achieved enough of its objective for assigned units to consider other missions, but a satisfied mission does not necessarily end. This tag is used for mission triggers and for multi-mission priority.

“Running”:

A running mission will have its assigned mission to execute the mission.

“H-Hour”:

In Command, H-Hour designates the date and time at which the mission designated as the initial mission for H-Hour is to start. H-Hour in Command isn’t strictly an H-hour according military terminology as it can be customized without restriction.

“L-Hour”:

In Command, L-Hour designates the date and time at which the mission designated as the initial mission for L-Hour is to start. L-Hour in Command isn’t strictly an L-hour according military terminology as it can be customized without restriction.

Understanding the concept of dynamic (aka. multi-missioned) units

The operations planner provides the capability for a single unit to be assigned to multiple missions. Of course, the unit can only execute one mission at a time, but you can now prioritize which mission it should execute. Mission priority is set via the operations planner.

Let’s assume we have a scenario where a group of aircrafts is assigned to a patrol mission. These aircrafts should patrol an area from a given time and then start a strike mission against a group of tanks. Without the operations planner, the player would need to track the time and then manually switch each aircraft to a new mission, at the right moment. Thanks to the operations planner such behavior can be automated, and in a more complex environment, our units could even behave like a reactive AI, aware of the simulation at the strategic level.

IMPORTANT ! A unit without “Dynamic” Checked will be ignored by the operations planner dynamic assignment, if you want a unit to work with the operations planner’s feature you **MUST** designate it as a dynamic unit.

Notice that since we have toggle “Showing Multi-Mission” we are now allowed to assign multiple mission to a unit, they are still shown in the “Available units” list despite having an assigned mission.

This panel on the right side of the mission editor shows all missions assigned to this unit. The mission in green in the one currently executed by the unit. Missions don’t have an order of execution, but a priority, which is set in the operations planner, as we will see later.

The “Dynamic” checkbox allows the unit to be assigned to multiple missions, units are all unchecked by default to reflect default command behavior.

Below the mission list we can see the current status and phase of the mission. We will see later in the operations planner chapter as “Phase” is a new way of managing the mission dynamically and is closely related to multi-mission:

It is on this screen that you decide to add the unit to all of the missions you want it assigned to. At this point you don't have worry about priority or to select a current mission as the operations planner will manage this for you.

In above's example we see that Rafale B is assigned to both the "Air Superiority Patrol" and "Light Tanks Destruction" missions, and the active mission for this unit is the one in green: "Air Superiority Patrol." Depending on the configuration of the operation planer, this unit may automatically switch to the "Light Tanks Destruction" mission at some point in the future.

Don't worry if the multi-mission mechanism is not entirely clear to you yet, as the mission editor is only half the story. The next chapter on operations planner will explain the other half.

Interlude: "Mobile facility" vs "Ground unit" and Split / Merge ground units

One of the most significant capabilities brought by the operations planner, is the possibility to have units dynamically change mission. But not all types of units are eligible for this kind of behavior. As part of that, It is important to understand the core difference between:

Ground units as facility which is a legacy implementation where ground units are represented as a moving, multi-aimpoint "facility" (see this old post explaining this concept).

Mobile ground units, a new implementation, which works like others active units such as aircraft, ships etc. The first hold a group of units abstractly represented as "mounts", while the last is a fully simulated individual unit. The core difference that interests us here is that ground units as facility are transported as cargo which doesn't have an existence in the simulation until it has landed (and spawned).

This means that we can't assign or queue them a mission until they have landed, and you will not be able to achieve a fully autonomous behavior for your units, in this case.

Cargo operations are now enabled for active units, meaning that all these limitations are now removed. However, you must have the right methodology to achieve this.

Unless you have to achieve backward compatibility or if a database entry is missing, you **MUST** use ground units, **NOT** mobiles facilities, to use the operations planner at its fullest.

On the new cargo edition form, note the "type", at the moment, the database has more content for mobile facilities:

Ability to Split/Merge ground units (facilities)

Legacy ground units associated as "facilities" can now be rearranged through this new tool.

Select an eligible unit, such as a landed detachment, right click on it to bring the context menu and click on “Split unit”:

This brings you this menu, with the details of the detachment:

You can also break the detachment into individual units with a single click:

Two eligible units can be merged together:

The Operations Planner

The operations planner adds a new level of interaction between missions and allows units to be dynamically reassigned from one mission to another.

Since the operations planner is sometimes tied to a landing plan, we have here the concepts of H-Hour and L-Hour to indicate overall operational time. These can of course be ignored, or used in a different purpose:

On the top left, one can define the H-Hour and the L-Hour values.

The H-Hour box on the left is where we can define the date and time to start the H-Hour mission. The H-Hour mission is the initial mission in the operation. L-Hour box works identically to H-Hour but they are independent.

Once an H-Hour or L-Hour is hit the respective initial mission can no longer be changed.

The checkbox in the middle ties the H-Hour to the L-Hour, meaning that the time separation will be constant between H-Hour and L-Hour when you modify the time for either.

The spreadsheet in the center lists all the side’s missions. Most columns are informational. Generally, you will be dealing with 2 columns: Priority and Phase.

Mission priority

The priority of a mission does NOT designate his supposed position in a mission execution queue. It indicates to its assigned units how important this mission is at this moment. The mission priority is used by units assigned to multiple missions to decide which mission to execute at any given time.

A unit having a mission in “On Hold” phase will not have this mission evaluated for the active mission evaluation.

Mission phases

A mission phase is a new concept introduced with the operations planner extension, it is not related to the mission status and should not be mistaken with it:

Waiting for trigger: The mission may (or may not) have already started yet but queued units won't evaluate this mission when choosing one to be assigned to.

Satisfied: The mission is considered to have achieved enough of its objective for its current assigned dynamic units to consider other missions, but a satisfied mission does not mean its ending. This tag is used for mission triggers and for multi-mission priority.

Triggered : A running mission will become a valid candidate for multi-mission units assigned to it. Therefore, a queued unit to this mission may become assigned to it.

A unit assigned to multiple missions will pick the most appropriate mission to be assigned to, based on the priority and the phase of the mission. unless it is already active on a mission that is in “running” phase, or if all assigned missions are in “On Hold” phase.

A unit that is active in a mission in “Satisfied” phase will still evaluate the satisfied mission and may continue that mission if no other missions are available.

Name

Just like for operation, the mission's name helps for organization. However user's made Lua scripts might reference a mission by its name and it is recommended to be careful when changing mission's name in such situation.

Description

This is purely an informational tag and does not affect the simulation at the moment. Use and edit this field to organize yourself.

Generated mission will usually contain some generated information

Type

This is the mission sub type as defined in Command's simulation.

Execution Time

The time, relative to H-Hour at which the mission is estimated to begin. See the chapter “Working with estimation” for more details.

Bulk actions

Bulk action tools are located on the bottom of the operations planner. These are used to select multiple mission at once and perform simultaneous modifications on them:

Filter

The filter tool is used to do a specific term research on mission and filter the result of this search query:

Triggers

This chapter is the core of the operations planner capabilities. It allows relationships between mission and a dynamic approach when executing missions.

A trigger is like a set of a condition and an action, if the condition is met, then we do an action.

Command evaluates all these triggers every second.

For mission, there are 2 types of actions:

We start a mission (we set the mission’s phase as “Running”).

We finish a mission (we set the mission’s phase as “Satisfied”).

It is important to understand that Command doesn’t have a concept of mission completion, when tagging a mission as satisfied, command only indicates that the mission have enough fulfilled its objective to allow its assigned units to evaluate other mission assignment options. Of course all relevant missions have already implicit mission completion mechanisms, a cargo mission will stop operation once the task is done, a strike mission won’t launch again to strike a nonexistent targets , etc...

These 2 actions are tied to a set of conditions.

Starting a mission with triggers

Select any of the mission and look on the right side of the operations planner window. Notice the main block called “Triggers to Start Mission”, and how it is separated into 3 smaller blocks:

These smaller blocks are individual conditions. Checking them means this specific condition will be evaluated.

The dropdown on the right of each smaller block is called a conditional operator.

As you can see there are 3 types of triggers that can start a missions:

A time based trigger

A mission dependency trigger

A Lua script trigger

Time-based trigger

This will be triggered when the scenario date reaches the defined H-Hour plus or minus a given duration.

Example #1 : This trigger will be true if we reach H+ 2 hours

Example #2 : This trigger will be true if we reach H- 23 hours

Mission dependency trigger

This will be triggered if all missions in “missions to check” are in “Satisfied” phase.

In this example, it will be triggered when “Air Superiority Patrol” mission is in “Satisfied” Phase:

Lua script trigger

This will be triggered if the Lua script contained returns TRUE as a value:

“Finishing” a mission with triggers

It was mentioned earlier that Command does not have an explicit concept of finished mission.

The triggers to tag a mission as “satisfied” work just like the one to start it.

The first trigger is a time based one, it tracks the elapsed time since the mission's phase has been set to "Running" and will be true once the defined time elapsed.

The second one is a Lua script trigger and work identically to the one in the mission start trigger – it returns the Boolean value of the contained lua script.

Logical Operators

Triggers, in Command, can be tied with logical operators.

Take a look at the picture on the right, representing a set of triggers to start a mission:

We have set all 3 triggers to be checked. On the right of each triggers you can notice a dropdown when you can selected either the OR or AND operator.

For the triggers to return TRUE and therefore start the mission (set its phase to "Running") each checked trigger is evaluated.

We see in this example that the first trigger has "OR" operator, the second "AND," the third "OR."

What it means in this situation is that the second trigger (the one with "AND") must be true.

In addition the "AND" trigger needing to be true, either of the first or third "OR" triggers must also be true.

If all conditions are met and the current mission's phases is "On Hold" then the mission will change it phase to "Running".

Another example (left):

All triggers are checked (and will be evaluated), and each of them have the "AND" operator.

This means that the missions will start when all conditions are true.

Operations planner and Lua Scripting

Command has already a powerful Lua scripting framework. The operations planner allows the integration of your script as a trigger.

Clicking on "Edit Script" in either the "Triggers to Start Mission" or the "Triggers to Tag Mission as Satisfied" group will bring you to an interface where you can add your script in:

Just like the rest of the triggers, the Lua Script trigger will be executed each second for its associated mission. The Lua script must return a Boolean.

Working with estimation

Command is such a complex simulation that giving an accurate estimation of an operation duration could take up to hours of computation. The operations planner provides an instantaneous estimation at the price of reliability.

The operations planner estimation takes into account all the time based triggers and mission dependencies, it simulates a run and then display the estimated execution time for each mission.

This estimation work only thanks to the user's input on triggers.

The special case of Lua script

You may have noticed that not all triggers are time-based, some depends on lua script and cannot be reliably predicted. In this case, you will have to manually input a value into the trigger in this trigger, shown on the right.

It is not necessary to check (enable) the trigger, having an unchecked "Time Elapsed" trigger with a value basically tells the operations planner : "Only use this trigger when estimating execution time". In this example, we assume the mission will be satisfied after 1 hour:

If everything is properly configured, clicking on the "Simulate" button will calculate the execution time, relative to H-Hour for each mission:

The Multi-Domain Strike Planner <https://command.matrixgames.com/?p=5492>

Throw away your planning spreadsheets! You asked/begged/hostaged family members for it, and now it's here. Coordinate massive, complex strike missions with time-on-target, complex flight plans (incl. in-flight refueling) , multiple attack patterns and multi-domain strike combinations. "Bringing everything together on a strike is just too complex/difficult" is officially over as an excuse. If you don't master this, your adversary most definitely will.

Poolside party: Task Pools

You have 8 bases across your scenarios, in every base there are 5 B-52 Stratofortress, and you want to assign a specific set of missions to a specific subgroup of those Buffs. Sounds painful?

No more!

With the brand-new Task Pool feature you can now easily create logical clusters of units and be able to operate them effortlessly in a large scenario like never before.

From the new and improved Mission Editor (F11)

Select Add,

from the popup select Task Pool,

then assign a name to the Task Pool just created:

You will now see the Task Pool just created (light blue circular symbol), the list of available units (round shaped green rectangle) and the list of units assigned to the Task Pool (orange rectangle).

Press the Arrow shaped buttons to manage the units into the Task Pool:

4 B-52s are added to the Task Pool just created:

This will let you quickly assign one or more Packages to those aircraft, allowing you to rapidly access to this specific set of units without having to search for them in your scenario.

This feature can be viewed like a more nuanced and detailed version of “Control Groups”, that are available in a large quantity of modern RTS. Task Pools let you create a cluster of units that are not forced to work together in any way, giving you a new way of interacting with multiple units at the same time.

Going the extra mile: Packages

The concept of Packages is deeply connected to the concept of Task Pool. Once you create a Task Pool, you can define one or more Packages to be executed exclusively by those units.

This link is immediately defined when you create a Package. In Mission Editor, click on Add and select Packages. Here, you'll need to define a Name and a Parent Pool, like in the image below:

You will now see the Package just created as a child of the Task Pool (light blue circular symbol), the list of units that are in the parent Task Pool, (round shaped green rectangle) and the list of units assigned to the Package (orange rectangle):

As before, you can press the Arrow-shaped buttons to manage the units into the Package.

You can interact with the Packages just like you would in a normal Mission. Expect every functionality to be the same, and every setting to work exactly in the same way. The only limitation would be that only units from the

parent Task Pool can be assigned to the target Package. The Task Pool is acting like a filter, lifting the encumbrance of having to look for that specific subset of units every time you need it.

30 mins or the next one's free: Time On Target (TOT) and Take/Off Time

Do you need a specific place to be blown exactly at 18:00:00 ?

We got you covered!

With the new TOT feature you will be able to schedule precise multi-domain Package in minute detail.

After creating a Strike Mission or Package, added the desired units and defined a target, you can now define a Time On Target (TOT) (light blue circle); then, click on CREATE or UPDATE flightplans (green rounded square):

The simulation has now created a detailed Flightplan for all the units involved in the Mission, taking into account the inserted time constraint. This Flightplans are visible from the Flightplan Editor Accessible from the appropriate button (orange square).

Let's analyse the Flightplan Editor in detail:

In the light blue circle, you can see the Flight name.

On the map, the Flightplan is visible as shown into the orange square.

Every Waypoint is shown in complete details, and you can examine and change type, time and speed of every Waypoint, as long as the physical constraints are respected.

You can also add or move most of the Waypoint. Every change is reflected in real time on the game map and vice versa, giving you complete freedom when customizing your Flightplans.

The TOT constraint is a multi-domain element that lets you precisely synchronize complex multi-domain strikes: Just add another type of unit to the mission with a TOT assigned, and every unit will hold its weapon until the firing time comes. The firing time is constantly re-evaluated and takes into account if the target or the firing unit moves.

In the example below, the aircraft has taken off at the expected Take-Off Time and will arrive on target at the expected TOT, but the ship did not fire yet:

The ship fired its weapon according to the weapon ETA:

and the weapons reached the target at the common expected TOT:

Seven sim-seconds after this screenshot was taken, both the string of iron bombs (notice the aircraft that released them is already breaking clear) and the ship-launched cruise missiles impacted concurrently on their targets.

Any which way you can: Attack Methods

You can define a wide array of attack patterns for your air strike, including – but not limited to – Multi staged TOT with various degrees of separation, single axis and split action. This will let you further customize your strike plans without the need to manually define those attack manoeuvres every time:

As with the flight planner, every Waypoint can still be dragged and customized as seen fit.

A sip in mid-air: Air-to-Air Refuel (AAR) Scheduling

This is a more advanced topic and one of the most requested features.

In the screenshot below, a Task Pool and a Package were created alongside a Support Mission:

From the Flightplan Editor of the Strike Package, Waypoint Number 7 (on the Egress leg of the flight) was selected and a new Waypoint was added. The added Waypoint (generated as the 8th one of the route) default type is “turning point” but it was redefined as a refuel point:

As soon as the Waypoint 8 is reached, the aircraft that are part of the flight will head toward the tanker and be refueled.

Cargo 2.0 – The main points <https://command.matrixgames.com/?p=5496>

Command’s existing cargo system was hitherto geared more towards the transfer of combat forces & personnel rather than materiel. This changes radically with Cargo 2.0. You can now transfer both combat units and also weapons, stores, fuel and any arbitrary material. Place your cargo on a multitude of different container types, from standard ISO-blocks to specialized boxes, each with its own peculiarities. Transload cargo at airbases, ports etc. in order to haul it over even transcontinental distances. Automate all this through cargo and (new) transfer missions. Set up complex logistical chains from mainland factories all the way to the front line. Expeditionary commanders from Napoleon to Alexander and Eisenhower would have given their right arm for such a tool – and it’s now included in Command for free.

Mission Editor / Mission Behavior Changes

(New UI elements circled in red)

A new cargo-oriented mission type has been added and the original type of cargo mission has been renamed. The existing cargo mission behavior is now called a 'Delivery' cargo mission. The new cargo mission type is a 'Transfer' cargo mission.

Delivery cargo missions work as before, with the destination being a zone defined by RPs on the map. This type of mission will unload cargo into action / for use in the simulation (units will unload from cargo onto the map where they can be given orders, etc. Ammo and fuel will move into unit fuel records / magazines.)

Transfer cargo missions, on the other hand, are for moving cargo from one holding unit to another –from airbase to airbase, port to port, or supply facility to airbase, etc. The cargo is not unloaded onto the map but instead moved from the starting cargo source into one of the transporting units assigned to the mission, transported to the destination, and then moved from the transporting unit into destination unit's cargo.

Creating a Transfer cargo mission uses the same procedure as creating a ferry mission – select a destination unit (rather than a group of reference points) before you create the new mission.

Ships and aircraft transfer or deliver cargo from their starting host unit. Ground units assigned to a cargo missions should start the scenario loaded as cargo within the source unit (i.e. a ship, fixed facility, airbase, etc.) They will automatically exit cargo and appear on the map as soon as the mission activates and they are within range of the mission destination.

Move All Cargo From All Available Sources: This is a new option for any cargo mission to move all the cargo from the available source(s). You can click this checkbox instead of having to manually go through and assign every possible cargo item. It also allows cargo to 'flow' through a system of cargo missions without the sim needing to know what cargo is / is not going to be transferred to a cargo source by some other cargo mission.

Vehicles Stored in Cargo May Self-Transfer: This is a new option found on the 'Transfer To' tab for cargo transfer missions. Setting this option 'on' allows ground units (not mobile facilities or other cargo) that are assigned to the cargo mission to transport themselves from the starting cargo source to the destination unit's cargo. They will exit cargo onto the map, travel to the destination unit, and then enter the cargo of the destination unit. In order for this to work the ground unit needs to be in the mission source's cargo, assigned to the mission, AND in the list of cargo to be transported (or you have the 'move all' option set to on.)

Cargo / Edit Cargo Changes

A new cargo type, 'container', is now available (NOTE: DB v493 or later is required). Cargo containers are added/removed from cargo the same as other types of cargo. Once a container has been added to cargo you can select it from the current cargo list on the left and click 'Edit Container' to put other cargo inside the selected container.

Cargo containers can contain ammunition, fuel, or user-defined contents. User-defined contents can be assigned a name, size, and mass via the Edit Container window. Containers are loaded and moved in the same manner as other cargo types but cannot be unloaded directly to the map as independent entities – they are always in cargo.

If cargo containers are moved as part of a cargo delivery mission they will be delivered into nearby (within 2nm) existing supply-type facilities if possible, or a new ‘forward arming and refueling point’ facility will be created to hold the containers if none are available.

Fuel in cargo containers that is delivered via a cargo delivery mission will be added as available fuel of the destination facility. This fuel can then be used by other units to refuel.

Ammunition in cargo containers that is delivered via a cargo delivery mission will go into the magazine of the destination facility. These can be used to rearm other units that use the same ammunition type (NOTE: Check the DB ID number to confirm your cargo ammunition matches the type used by the unit you want to re-supply).

User-Defined Cargo: This is a catch-all type of cargo for items the user wishes to move and track as cargo but which have no effect within the Command simulation. User-defined cargo can be used to track the movement and delivery of items like food, medical supplies, spare parts, etc.

Chaining multiple Cargo Missions

A single cargo mission can handle movement of cargo from the assigned source(s) to one destination. By using the ‘Move All Cargo’ option, however, you can create a sequence of cargo missions to ultimately move cargo through a series of intermediate destinations and on to delivery in the field – or in an extreme example, move material from homeland factories all the way to the expeditionary frontline. The ‘Move All Cargo’ option is required for this sort of system, as individual cargo missions do not intrinsically know what cargo may be delivered to their cargo source(s) by other cargo missions.

The ability to use multiple cargo missions to route different cargo items from the same source to different destinations is currently limited to ‘originating’ cargo sources – i.e. those where you know no other cargo mission will be delivering cargo TO the cargo source.

New Game Option: Show US Units of Measure in Cargo Editor

This option will change the display values show in the Edit Cargo and Edit Container forms to use US units of measure (feet / square feet / tons / pounds / gallons) rather than metric units. This option applies ONLY to those two forms. The forms will refresh automatically if change the option.

Simulation Additions & Improvements <https://command.matrixgames.com/?p=5500>

NEW FEATURE: Energy-based flight model for boost-coast missiles

Boost-coast anti-air missiles (ie. most tactical missiles that are not powered continuously) now use a much more realistic flight model that distinctly models the initial boost-sustain and post-burnout regimes, and takes into account the effects of gravity (shedding speed while climbing and regaining it when diving) and aerodynamic drag. The drag changes with altitude, built-in drag coefficient and whether the weapon is maneuvering (pitching/turning) or not. This change makes it possible to apply real-life “exhaust the threat” tactics and further constrains edge-of-envelope shots.

The onboard fuel (and thus boost duration) varies with the type of missile propulsion. Most AAW missiles (e.g. Sidewinder, Sparrow, all Standards, pre-D AMRAAMs etc.) still rely on the “traditional” boost-(optional sustain)-coast sequence, in which case the rocket motor is active usually for a few seconds. Some missiles (SA-4, SA-6, Sea Dart, Meteor etc.) use ramjet propulsion to provide for a much longer burn duration, and this allows them both a much higher average speed-to-target but also a higher energy state on the terminal engagement, which increases their chance of impact. Other new systems like the AIM-120D use “dual-pulse” rocket motors to again achieve a substantially higher overall energy state.

(NOTE: On missiles that use this model, the “fuel bar” indicator now represents only the remaining boost-sustain fuel, NOT to the total remaining energy. After burnout, the fuel bar is removed and the weapon will coast until it reaches its stall speed.)

Significant changes in default aircraft defensive maneuvers

Instead of beaming and diving to the deck by default, now they will first try to outrun an incoming missile while matching its relative pitch (i.e. climb if the missile is below them, or dive if it's above them), and if the missile closes the distance they will then attempt to beam it (or its parent guidance) while also reversing their climb/dive.

To counter these counters, new additional WRA firing-range settings (including “No-Escape Zone”) are available, offering a much more comprehensive set of range options (see the UI improvements article). A2A and S2A missile engagements are, as a result, both more dynamic and far more realistic now.

(NOTE: These two changes have been arguably the most controversial ones during the public beta of the War Planner. The typical complaint by many players is “My AMRAAMs are now useless unless if fire them almost at point-blank range”. Our response to this is: EXACTLY. Welcome to the real-world kinematic limitations of most AAW missiles (against agile & alert targets, at least). This part of the reason that most (all?) real-life BVR kills have been achieved at significantly less-than-nominal launch ranges. Watch this BVR tactics video from F4 BMS and note how on each case the missiles are dragged-out rather than outmaneuvered. WRAs and configurable firing ranges are a thing – and with the new percentage-based settings and NEZ they are more powerful than ever. Learn them, practice with them and use them. Or get used to becoming your adversary's chew-toy, first by the enemy AI and later by other human players as MP comes to commercial CMO.)

NEW FEATURE: Passive Coherent Location System (aka “Passive Radar”)

The term “passive radar” has grown to broadly encompass two significantly different concepts; one is ESM/SIGINT-based air-surveillance systems and the other is multistatic radars with third-party emitters. The

latter is called Passive Coherent Location System, and for a general background on the technology and CONOPS see here: https://en.wikipedia.org/wiki/Passive_radar

PCLS systems can be very capable against VLO targets (if the geometry is right), and their passive nature makes them inherently less vulnerable to SEAD attacks (although of course the emitters-of opportunity may themselves be targeted). On the other hand the geometry restrictions can be a tough taskmaster (the emitter, the receiver and the target must form a clean-LOS triangle otherwise there is no detection) and the signal propagation geometry sharply reduces the effective detection altitude. Therefore such systems are most effective when combined with other traditional active & passive surveillance sets rather than operating completely on their own.

The v488+ releases of the DB3000 database contain several “non-detecting emitter” platforms such as radio, TV, and cellular antennas or navigation beacons such as LORAN. It also contains a prototype PCLS vehicle that can be used in testing and as the basis for operational variants.

NEW FEATURE: Distinct mobile ground units

In addition to modelling mobile forces as “aimpoint facilities” (see: <https://www.warfaresims.com/?p=1159>), it is now possible to explicitly model individual vehicles with their own customized properties such as armor, propulsion, mounts, sensors etc.

The new-style ground units have unlocked certain brand-new capabilities, such as true amphibious vehicles (with distinct speed & fuel consumption properties overwater and on land).

NEW FEATURE: Intermittent emissions

This band-new feature allows to control the behavior of emitting sensors so they emit in intervals instead of only continuously or never.

Alert levels

The first thing to do is to configure a unit’s interval according to the alert level. The alert level is set in the EMCON setting of any unit and affects the whole side. Below, in the “Active Emission Interval” we have the intermittent emission configurations for each alert levels. The side’s alert level will determine which intermittent emission configuration to use:

Emission Intervals

Let’s configure an EMCON behavior for the yellow’s alert level. By default, the interval type is set to continuous, this is the usual behavior in command where an enabled sensor will emit continuously. Emission Duration is the duration in seconds of an emitting phase. Interval is the duration in seconds of a silent phase between 2 emitting phases. Interval random variation is the random duration in seconds we add to an interval. This allows unpredictability of the cycle and is particularly useful to disrupt the enemy’s plan:

The action of waking up mean that the radar will temporarily turn into continuous and ignore intermittent behavior, until time until sleep mode is elapsed. The Wake when detecting threat checkbox controls this behavior, a threat is a unit defined by the checkboxes group includes stance and includes ID. If the radar detects an unfriendly or hostile or unknown contact with any identification level, then we wake up the radar.

If you want a unit to inherit all configuration from its parent check Use parent group parameters:

NOTE: intermittent emission will NOT make a radar active or inactive as depicted below:

Intermittent emission controls whether or not an active radar will be silent or emitting at a given moment.

Custom emission intervals configuration

If you want to use all alert level, you will need to define each interval configurations, and perhaps you want a way to override the alert level and have some units feel special and use their own rules:

To do this, populate the Custom interval tab as you will and make sure the Use custom preset only checkbox is checked.

NEW FEATURE: Custom Environment Zones

Multiple & moving weather fronts? Check. Bend the laws of physics on a localized area? Can do. Specify carefully hand-picked weather, terrain and other environmental properties in order to test or compare sensors and other environment-dependent components? Yup. Unleash your inner nature wizard with this puppy.

Using this new feature, you can define a zone where you can tailor the environment & weather properties. This can be useful if you want a “controlled environment” for sensor checks, mobility & damage tests etc., but can also be used as a localized “weather override” for scenario purposes.

To create a CEZ, bring up the Refpoint Manager and switch to the “Cust Env Zones” tab. Create a zone as usual, and then click on “Edit”. A new window should appear, in which you can define the weather & environment properties:

NEW FEATURE: Palletized weapons and other stores

This is a new capability that has been making the public rounds lately, as a result of a series of videos by AFRL on the Rapid Dragon concept. using pallets packed with guided weapons, aircraft not usually associated with frontline attack operations (such as transports) can contribute to the firepower volume allocated at enemy forces.

As usual, there are caveats. The fact that weapons are fired from released pallets, rather than individually fired from the parent platform, means that weapon allocations must happen in batches; if a single missile in say a 12-pack is allocated, the full dozen has to be allocated either on the same target or others. (There exists of course the theoretical option of allocating only the desired amount of weapons and just sacrificing the rest of the pack, but the cost of the majority of modern weapons makes this an unlikely scenario).

Therefore, accurately modelling this new capability (and the decisions & restrictions it enforces) to Command has been a lot of work. Here's what we've come up with:

Pallet Weapon: the paratroppable system (usually a pallet, but may also be a container etc.) containing the payload

Palletized Weapon: the content of the pallet, the actual weapon that will be delivered

Pallet Weapons can be fired by specific aircraft that are equipped with suitable loadouts (C-130, C-17 etc.).

Several new representative loadouts have been included in the DB3000 database:

Pallet and Weapon Allocation

Pallet Weapons can be allocated to a target both as a Pallet or by assigning a single target for every Palletized Weapon in the aircraft Loadout. When allocating a whole Pallet, all the weapons within it will be added to the salvo, and all of them will share the same target as the Pallet:

Single (individual) Palletized Weapons can also be allocated clicking on the relative node in the tree:

NOTE: As mentioned, due to the nature of the weapon system the whole Pallet is dropped even when some of the weapons within are not allocated. So take care to allocate the full pack! A message will serve as a reminder in order to avoid wasting weapons:

When the Palletized Weapons are allocated separately, the system will recognize how many Pallets are needed to fill the requested salvo quantity and will drop the appropriate amount of Pallets:

Pallet and Weapons Behavior

When dropped from an aircraft, the Pallet will align itself following the correct loitering pitch, and after reaching that pitch it will deploy the parachute and start to loiter. After the Pallet starts to loiter, all the allocated weapons are fired from the Pallet:

Example with multiple Pallets:

Pallets have one more tick up their sleeve: Because they remain in coms (datalink) with their parent aircraft, you can allocate additional weapons remaining in them while they are para-dropping. Obviously this can be very useful for delayed fires against targets as the tactical situation evolves. This can be done simply by clicking on a Pallet and then allocating its hosted weapon(s) as usual:

ZueeFXi.png (1017×609)

Effect on WRAs

All the WRA will now include the weapon info for the Palletized Weapons. Since the Pallet itself is just a carrier, it is not subjected to any WRA and will follow the WRA of its hosted Palletized Weapons:

New facility category: Surface + Underground

This new facility category represents facilities that are buried underground but also have major access from the surface in order to operate. Examples are all ballistic missile silos, some command bunkers, retractable coastal-defense turrets like ERSTA etc. These facilities are vulnerable to damage/destruction both from direct weapon impacts on their surface area and also from underground shock from near misses by penetrator weapons (or exceptionally powerful surface detonations e.g. from a multi-megaton warhead or asteroid impact).

Significant improvements on ballistic missile kinematics

Ballistic trajectories have been reworked from the ground up, using true Kepler equations to reflect the movement of planetary bodies. This produces true-to-life boost-phase profiles and overall trajectory parameters (this is important in order to more faithfully model the abilities & limitations of BMD systems).

Improvements in ABM DLZ calculations

ABM systems have additional fail conditions in their DLZ evaluations compared to normal SAMs. A prime example is the intercept “hard floor” for exo-atmospheric systems. For instance, SM-3 cannot make intercepts under 100km in altitude (because its “warhead”/kill-vehicle is effectively a miniature spacecraft with a very sensitive IR seeker and no aerodynamic control, and thus cannot function within the atmosphere). This factor severely restricts the system’s intercept window: If the estimated intercept point is within the atmosphere, it is already too late to shoot. This screengrab from a LM THAAD-ER promo video illustrates this well:

In addition, the case of “intercept will happen within weapon minimum range” has been added as a fail condition to DLZ checks.

Overhauled Reaction Times

The differences in reaction times, and their effects, are now more critical than ever. All units use common-reference “Combat System Generation” (“Cockpit Generation” for aircraft) to model the modernity of their combat systems, combined with an “Ergonomics” value to handle intra-generation differences (the atrocious switchology on early missile-age aircraft will most definitely get you killed now). Older, WW2-era ships may take up to 5 minutes to engage a target, while Aegis cruisers fire in <20 seconds. Cold War fighters will be beaten to the draw by modern, fifth-generation fighters. Overmatch, that ever-elusive dream, is now possible – but beware, it goes both ways.

Until now, the existing OODA model was hobbled by two major problems. First, most values were way too fast. Detection, targeting, and/or evasion usually took less than 20 seconds. This works for ships equipped with modern combat systems, e.g. upgraded Aegis; not so for Cold War clunkers (or, heaven forbid, a pre-WW2 battleship). Second, having to manually set OODA values per-ship meant inconsistent reaction times across combat system generations.

For the revised model, we’ve added a new component to ships: the Combat System Generation (CS Gen), which can range from Gen 1 (1945-1950) to future Gen 7 (2030+). Whether you’re plotting contacts on a plotting board, federated CPUs, or an integrated system – and the age of the components making up those systems – will have a huge effect on your ability to quickly react to and engage targets. A modern Aegis ship can spot, track, and engage a contact in about 20 seconds; for a ship with a WW2-vintage CIC, the process balloons upwards to almost five minutes. If your first detection of an incoming missile comes from your own radars, it may not be enough.

There is an exception for small craft, which regardless of age generally have extremely fast reaction times (this is because their targeting process essentially consists of shouting at the guy on the MG to “shoot that guy over there RIGHT NOW”). However, this is mitigated by their comparatively less powerful sensors and weapons: being quick on the draw doesn’t help much when the Mk1 Eyeball on your RHIB spots a Hellfire inbound.

“But wait,” we hear you object. “Doesn’t this mean that, given an especially old ship and a modern threat, I may find myself completely helpless against an incoming missile?” The answer is yes: those Final Countdown scenarios are going to play out very differently now. This was a major concern for naval planners during the Cold War (the development of NTDS, the US Navy’s first automatic data-exchange system, was driven by classified exercises & wargames in which USN ships were consistently sunk before they could react to incoming air & missile strikes).

For aircraft, the flying equivalent to the “Combat System Generation” for ships, the “Cockpit Generation” is a way for us to approximate pilot load – and subsequently reaction times – based on cockpit design. Aircraft can be assigned one of six cockpits: “Basic Instruments Only,” intended for the simplest aircraft; “Steam Gauges,” as for WW2-era fighters; “Complex Steam Gauges,” for the nightmarish mid-Cold War fighters; “Partial Glass Cockpit,” for transitioning aircraft of the ‘70s-90s; “Glass Cockpit,” for most modern aircraft; or finally a

“Panoramic Cockpit Display,” representing the new single-screen touch displays coming to a fifth-generation fighter near you.

Just as a ship is only as good as its CIC, your aircraft are only as good as their cockpits – and whether your pilots are forced to fiddle with a forest of knobs, switches, and dials or able to glance at an easy-to-read LCD will affect their performance. Of course, the difference between aircraft cockpit generations is far less than the difference between ship CS generations, measured in seconds rather than minutes – but in an aerial engagement, seconds do count.

For submarines, facilities and mobile ground units, we have followed a similar path. We split submarine combat systems into six distinct “generations,” ranging from the early Mk1/3/4 TDC used in WW2 to the modern AN/BYG-1 and future witchcraft as featured in SSN(X). Facilities and ground units required a slightly different system: rather than using multiple “generations” as with ships and subs, these annexes distinguish between fixed and mobile systems and whether said system uses manual or assisted guidance. For example, radar-assisted AAA (e.g. Skyshield) will be faster to engage than a manual gun (ZSU-57); both (towed) systems will be slower at evasion than a SPAAG. The same is true for manually laid artillery vs. those with digital fire control, towed artillery vs. SPGs, etc. While less detailed than our system for ships, subs, and aircraft, we felt this was “good enough” for the level at which these platforms are simulated in Command. (And, of course, we always have the manual override for when we have specific platform data.)

This is obviously is a major set of changes and potentially game-breaking for older scenarios. For this reason, the SBR tool now also includes the ability to preserve “legacy” OODA values when migrating scenarios to v494+ databases. For pro users, the DB Editor also offers the ability to explicitly set custom values and selectively override the generation-derived ones.

Extra wrinkles: Ergonomics

Not every aircraft built with steam gauges was equally difficult to fly; not every ship built in a certain decade could react with identical speed. Certain platforms gained a reputation for being especially easy to operate (the Viggen, for example, had a remarkably well-designed cockpit); others, like the Komar-class missile boat, were the bane of their operators. Much of this can be ascribed to ergonomics, the consideration (or lack thereof) of human factors in design.

The new “ergonomics” field – ranging from “Awful” to “Excellent” – is intended to reflect these intra-generation differences, acting as a sort of OODA “buff/debuff” and giving the ability to adjust values to reflect upgrades, etc. without needing to take the drastic step of upgrading the combat generation.

And, of course, you still have proficiency in the mix. Players now have an interesting mix of factors to consider:

- “Tech generation,” e.g. CS/cockpit generation: when was your platform designed/built, and what sort of tech was in play at the time? Are you working with ancient WW2 plotting boards or Aegis?
- Usability/ergonomics/design: are you working with beautiful, top-of-the-line COTS human interface tech or the nightmare that was mid/late-Cold War Soviet systems?
- Proficiency: Do you have well-trained, well-rested/motivated crews, or are you dealing with bottom-of-the-barrel conscripts?

Those three factors will all have to play into your operational planning.

Radar & IR Stealth Modifier Improvements

Sensor improvements come coupled with a massive overhaul of signature modifiers in the DB, which significantly improve the realism of our stealth model by drawing clearer distinctions between shaping and RAM generations.

Prior to the v494 DB releases, we could classify an aircraft as having “light,” “medium,” or “heavy” stealth shaping ... and that was it. These modifiers were applied always in all aspects (no ability to define e.g. a frontal-only reduction modifier). Modeling of IR signature suppression (IRSS) techniques was even more limited.

In v494+ we completely overhauled our existing VLO modifiers to account for shaping and RAM generations. In addition we also added several special flags to indicate the presence (or lack thereof) of certain stealthy design features. This allows us to model not only general, whole-craft stealth but also context-specific or aspect-specific features such as S-shaped intakes, exposed fan blockers, active cancellation, and stealth pylons. For example, S-shaped intakes reduce the likelihood of being detected head-on, while LO pylons reduce the impact of externally-carried stores.

This overhaul also extends to IR modifiers. As with radar stealth, we completely rewrote our “general” modifiers to represent whole-aircraft IR signature suppression techniques (distributed vs. conventional fuel tanks, low-E coatings etc.) and added several additional aircraft codes to represent specific IRSS features. These codes include shielded “anti-Strela” exhausts, masked exhausts, heavily masked / slit-shaped exhausts, and peak temperature reduction or “cool-air mixing”. Note that certain IRSS features come with downsides and limitations: slit-shaped exhausts, for example, will make you harder to spot but paradoxically easier to lock on to with IR weapons due to back pressure penalty; in another example, anti-Strela exhausts are most effective against someone trying to get a lock from below.

The full list of added signature modifiers is:

RCSS – Active Cancellation

RCSS – S-Shaped Intake(s)

RCSS – Exposed Fan Blocker(s)

RCSS – Stealth Pylons

IRSS – Shielded Exhaust (Anti-Strela)

IRSS – Masked Exhaust

IRSS – Heavily Masked / Slit-Shaped Exhaust

IRSS – Peak Temp Reduction (Cool-Air Mix)

We’ve backfilled all LO/VLO aircraft in both DBs with these features as best as we could determine. (Naturally, in many cases and especially with contemporary stealth fighters, exact details are sometimes hard to come by.) These changes mean that various LO/VLO aircraft are now much harder (or easier) to detect than you may be used to. We have solid confidence in the results; comparisons with known real-world RCS & IR data yield accurate numbers. However, we’re also open to feedback: expect tweaks in future DB releases as we hone the new values. Pro users can of course manually input their precise classified figures as before.

IR & Visual Sensor Improvements

Closely related to the above, this was initially presented and discussed in the professional edition context (see presentation side to the right) but applies equally to the commercial edition.

IRSTs and high-mag cameras are no longer near-magical counter-VLO sensors. They may still be your best bet for detection, but you won't be volume-scanning for stealth fighters at >100nm anymore. (You can still spot/track them pretty far enough IF something/someone else first cues you there). Abhirup Sengupta broadly explains the tech background behind this limitation well in this Quora post.

The relevant sensors now have a dual value in the search range listing in the DB value, to make it more explicit where their volume search extends to. For example, this listing for the PIRATE IRST system on RAF's Typhoon Tranche 3:

...indicates that the system can perform un-cued volume search out to 20nm, but can track already-detected targets out to 100nm (both cases under ideal conditions).

Visual and IR checks are now also susceptible to look-down clutter. For example, it is easier for an IRST (or the plain Mk1 Eyeball) to pick out an aircraft over the horizon line than against the surface background.

Degree-Definable Sensor Arcs and Vertical Scan Limits

This is a seemingly small but important improvement to our sensor modelling: at last, sensor arcs can optionally be defined in degrees, rather than just in "pie wedge" set sectors. We've also implemented vertical sensor arcs, which were especially important during the Cold War. Older air-to-air radars were often limited to a small chunk of vertical space (20 degrees or so), which meant that fighters would struggle to detect aircraft far below or above them. For air planners, this meant "Low CAPs" and "High CAPs" were necessary. Soon you too will have to consider altitude in your CAP coverage.

(Note that current DBs do not have vertical sensor arc data backfilled, so this won't have an immediate effect on gameplay until we can do a bit more research. Expect to see an RFI in the public Github for arcs and scan limits.)

Parallelized attack salvo calculations

The creation of attack salvos is now largely run in parallel instead of sequentially.

Some background to this: Until now, the creation of attack salvos was done purely sequentially. Unit-1 examines if it can attack (both doctrinally and physically) a contact; if yes, it creates a new salvo or adds its weapon(s) to an existing one for the same target with same weapon ID. Then Unit-2 does the same, etc. etc. until a suitable salvo for the given target is filled to capacity. This presents a number of issues:

Because of the sequential nature, a potential “Unit-3” which perhaps is better located to attack the same target, or perhaps has a more suitable weapon for it, may not be considered at all (if a suitable salvo for the given target is filled-to-capacity by previous units). This can lead to problematic situations like this. Again because of the sequential calculation, this process does not exploit multiple CPU cores and can significantly slow down a pulse execution in a large/busy scenario. Units now perform the salvo evaluations in their own individual threads and submit “firing proposals” to their side. The parent side groups firing proposals per-target and selects the most promising one, based on criteria that depend on the nature of the target (e.g. for aerospace targets an important factor is time-to-impact). Once selected, the salvos are executed as usual.

Weather effects for surface ship speed

This is an optional new feature. When enabled, deteriorating weather conditions (and especially increasing sea state) has an adverse impact on the maximum speed that ships can travel. This effect is particularly acute on small-displacement ships. Depending on sea state and ship size, a ship may be forced to run at 3/4, half, 1/4 speed or even heave (effectively remain stationary).

The information about the weather-related limitation is shown in various ways:

On the ship datablock

On the Unit Status panel

On the throttle/altitude window

Because this feature can potentially unbalance existing scenarios, it is optional (can be toggled on the “Scenario Realism Features” window) and is “ON” by default when making a new scenario, and “OFF” by default when loading an existing scenario.

Aircraft maximum airborne endurance

This fixes the “aircraft may stay up indefinitely by multiple A2A-refuellings” realism flaw. Aircraft are now limited in their total airborne endurance depending on their size, type and crew complement. The information about current airborne total time and maximum endurance is listed on the fuel panel and is color-coded for at-a-glance evaluation (dark red is bad). If an aircraft reaches its max endurance limit, it enters an “RTB – Exhaustion” state, turns straight for its home base and will refuse any manual orders to change course or engage in any other activity.

Other simulation bits and pieces

– Added new ASuW doctrine option: “Use missile waypoints” (NO by default). When set to Yes, ASuW missile shooters will use the missile’s waypoint/dogleg capability (if present and if the range allows it). If set to No, ASuW missile shots will always go “straight in” (for old-timers, this is the default CMANO pre-v1.10 behavior).

– Non-ABM-optimized AAW missiles (e.g. Stunner) use a direct-flight profile when engaging ballistic targets (even if they have a lofting/cruise altitude set in the DB)

- An AAW missile's attitude control type is now taken into account when determining Ph reduction due to reduced terminal energy:

Interceptors with non-aerodynamic control (e.g. SM3, THAAD) have no reduction at all

Interceptors with aerodynamic-only control (ie. most SAMs) have the as-standard reduction

Interceptors with combined aero & non-aero control (e.g. Aster 15/30, 9M96) halve their Ph reduction (the non-aero control diminishes the negative effect of low terminal energy on maneuverability)

- Optional speed improvement: Do not share-and-deduplicate contact messages between allied sides.

HOW TO ENABLE: On Command.ini, under the category [Game Preferences], add this:

"DedupAlliedMessages = False"

PROS: In scenarios with numerous sides allied to each other, this can provide a significant boost to sim performance.

CONS: When this is enabled, sides do not receive contact-related messages (new contact, contact status change, contact lost etc.) from their allied sides. NOTE: The contact data itself is shared as always, it's only the log-messages that are not shared. This may or may not be a significant hindrance depending on your gameplay style.

- Ships & submarines now attempt to evade incoming torpedoes more realistically, following these guidelines. Submarines will additionally alter their depth to avoid the torpedo(es) if appropriate.

- Added new sonar sub-type: Vertical Flank Array (DB v494+ required). These have been present in some Russian sub classes for a while, and are now backfitted on the Ohio class and earmarked for future US SSNs. They provide improved tracking capability (can establish a high-quality TMA track sooner).

- Expendable UAVs (and weapons/decoys) can now have their plotted course programmable in throttle & altitude

- Surface targets are now affected also by underground explosions, though not nearly as much as underground facilities (they receive much less of the transmitted shock). This makes it possible to damage/destroy buildings through near misses by penetrator weapons.

- New simulation feature: Certain weapons occupy their launcher while they are enroute (e.g. most wire-guided torpedoes & ATGMs). The mount cannot be reloaded before the weapon is destroyed one way or another. (Requires DBs v489+)

- ADDED: RWR detection of SARH/TVM illumination auto-generates an (incoming) missile contact, even if the missile itself has not yet been detected yet. This provides realistic ahead-warning (and thus evasion opportunity) to the targeted aircraft.

- New simulation feature: Daisy Chain-type weapon datalinks (DB v489+ required). This allows weapons to communicate directly with each other instead of being slaved to a datalink platform.

- The AI can now employ Mobile Air Decoys (MALD/TALD) (DB v489+ required).

- Support missions can now be created without a ref point (with a visual warning). Example use: Turn on ground based radars without having to turn them on the side-level or individually.

- AI improvement in wingman logic: Distances when evaluating contacts is measured from the group LEAD (not from itself), so the wingman will not chain-investigate away from lead. Also, wingmen do not investigate contacts alone if some other aircraft in the same mission is checking it. So if there are more unknowns than flights, wingmen may start breaking off to investigate them when they are in weapons range of their lead, otherwise they will not.
- Dual-seeker ARMs (e.g. AARGM) can now be fired (non-BOL) against non-emitting targets
- Relaxed the “have reached formation station” check threshold for aircraft (makes them less eager to throttle up to get to station)
- Ship attack AI tweak: When attacking with a contact bomb (ie. saboteur or kamikaze), if possible use flank throttle to minimize the target’s reaction window.
- The visual & IR signature modifiers of in-flight missiles have been tweaked so that they are picked up at shorter ranges. This prevents unrealistic early detection of incoming missiles and thus too-effective maneuvering against them.
- Aircraft AI tweak: When in normal loiter mode (no mission flight or plotted course), wingmen ignore their formation station and just stick close to the leader
- Added: Fuel resupply and rearm for submarines
- Tweaks to sonar cavitation model:
Cavitation is now dependent on desired speed (ie. screw turn rate) rather than current vessel speed
The noise boost from the onset of cavitation is now gradual (instead of all or nothing), and its magnitude depends on how much above the cavitation threshold the desired speed is
- Aircraft will now pitch up/down as necessary if weapon release is blocked by vertical boresight limits
- When undetected units fire gun/missile weapons, their visual signature blooms temporarily and they are susceptible to detection.
- BOL-fired torpedoes will snake if they are able to
- Added new navigation doctrine option: “Direct path” for ground units. This causes land units to head straight towards their destination without any pathfinding considerations.
- Added: Mine-Clearing mission can now also use “repeatable loop” movement style (so e.g. use explicit sweeping lanes)
- MIRVs are now released from ballistic missiles in sequence, one every 5 seconds (instead of all together)
- ARMs are less effective against low-band emitters (increased chance of a miss)
- Active sensors on aircraft are temporarily de-activated when approaching a target tanker for refueling (this is a safety measure to avoid igniting the tanker’s fuel).

- ALARM parachute loiter capability added (this requires DB3000 v489+)
- Vehicles now use cargo load/unload time calculation similar to aircraft
- Cargo transfers now attempt to keep grouped units (that are stored in cargo) together when possible.
- Amphibious vehicles assigned to an active mission but stored in cargo will be checked against available docking facilities on their host unit. If available, they will self-unload into the docking facility. This is currently checked every 30 seconds though only one unit can self-unload per check. Amphibious vehicles cannot use davit type docking facilities.
- Docked amphibious vehicles that are assigned to an active mission will launch automatically if they're in range of the mission area. This is the same behavior as boats currently perform for cargo missions, but amphibious vehicles auto-launch for all mission types.
- Ground Units can now perform cargo missions.
- Added "X days underway" tracking value for subs (example).
- Tweak to radar clutter logics: AESAs which refresh/revisit existing target tracks can narrow their beams and effectively cancel-out clutter altogether. This does NOT apply to volume-search scan against not-yet-detected targets.
- Implemented distinct horizon propagation values for different sensor types. Typical factors are 1.23, 1.5, and 1.06 for radar, ESM, and EO/IR respectively. These parameters are defined in the "Electronic Warfare and Radar Systems Engineering Handbook".

Weapon Release Authorization (WRA)

Command 1.07 adds a Weapon Release Authorization (WRA) tab to the Doctrine window. The WRA lists all weapons for the current side, mission, group or unit. To change the WRA for a weapon, press the (+) to expand the target list, and update the desired setting. Players and scenario designers can configure the number of weapons to use against a target and also the number of shooters that may participate in a salvo. Furthermore, the 'automatic firing range' of weapon systems can be capped so that the Artificial Intelligence (AI) won't automatically fire the weapon at certain target types beyond a given range. And lastly, each weapon system's self-defence behaviour can be configured in detail.

Weapon quantities

The WRA has two levels of targets: 'Unknown/Unspecified' and 'Specific' types. The first level is made up of 'Unknown Type' and 'Unspecified' target types, which are highlighted in bold text and used before and after a target has been identified. I.e. a blip on the screen uses 'Unknown Type' settings, and once identified as an aircraft or guided weapon the 'Unspecified' settings are used.

The WRA allows further refinement of the weapon allocations by specific target types. For example, the WRA can be configured to fire two missiles at a target identified as a MiG-29s and one missile at a Tu-95. Click on

the screenshot to the right for an example. Here, the RIM-66M-5 SM-2MR Blk IIIB SAM will be fired in pairs against 4th and 5th generation fighters and supersonic missiles, and singly against all other target types. It should be noted that in this screenshot, most settings are inherited from side level since they have not been configured locally, while some specific aircraft types have not been configured at all and inherit from the 'Unspecified' settings.

The WRA settings are inherited the same way as doctrine and EMCON are. The WRA can be configured at side, mission, group or unit level, and inherit settings from higher levels if not set at the current level. This makes the WRA very flexible, i.e. aircraft flying Mission A may be set up to fire 2 weapons at a given target, while all other units (from side level via missions and groups down to unit level) fire 4 of the given weapon at the same type of target.

When firing a SM-2MR Blk IIIB against a contact identified as a MiG-29, the simulator will first check the WRA setting for a 4th Generation Fighter/Attack aircraft from unit level via group and mission to side level. If not defined at any of these levels, the simulator will check the 'Unspecified' category from unit to side level. The 'Unspecified' target type is always specified at side level, and is the last place the WRA looks.

In addition to specifying the exact number of weapons to fire, anti-ship and land-attack weapons may also rely on the target's 'Missile Defence' value to automatically determine the number of weapons to fire depending on the target's defensive capabilities. See screenshot to the right for an example. The 'Missile Defence' value can be found in the Database Viewer (DB Viewer) for ships and facilities, and gives an indication on the number of Harpoon / SLAM / Maverick missiles needed to destroy the unit. The weapon may also be configured to use multiples of the 'Missile Defence' value. For instance, a less powerful weapon may be configured to use twice as many weapons, where as high-performance weapons like the AS-4 Kitchen or SS-N-19 Shipwreck are configured to use 1/4th the 'Missile Defence' value. In other words, a ship that is estimated to take 16x Harpoon missiles to sink will only have 4x AS-4s fired at it.

The WRA allows more refined 'Weapon Control Status' (WCS) than the FREE / TIGHT / HOLD settings in the Doctrine tab. Granted the WCS is set to FREE or TIGHT, the Artificial Intelligence (AI) may be excluded from automatically firing at certain targets to prevent employment of heavy anti-ship missiles against tiny boats that are best dealt with using other weapons such as guns. In the above screenshot the Harpoon missile has been configured to not be used against 0-500 ton vessels such as torpedo boats, smaller minehunters, open boats, tugs, fishing boats, small landing craft, etc. However the weapon will be used against 0-500 ton missile boats. Please note that this only applies if the target has been positively identified as a 0-500 ton boat. Otherwise, the AI will use the 'Unknown Type' settings and fire two weapons.

Number of shooters

In addition to limiting the number of weapons to fire, the number of shooters to participate in a salvo can be configured in the WRA. Typically, air targets will only have one unit firing at a time to avoid taking up more channels-of-fire than needed. Surface targets may have more shooters, for instance two ships participating with 8x Harpoon missiles each in a 16-round salvo against a Sovremenny DDG. Shooters will seek to fire all weapons in a salvo by themselves, and will only ask other units to join if there are not enough weapons available on the first unit.

Automatic Firing Range

In addition to control whether or not a weapon can be automatically fired by the Artificial Intelligence (AI), and preventing a weapon to automatically fire depending on target type, the WRA also allows the player and scenario designers to limit automatic fire by target range. For instance, a 80nm range weapon may be limited to automatic fire at 30nm. The player will of course be able to allocate weapons manually out to full weapons range.

A typical example would be to reduce the firing-range of SA-5 Gammon SAMs to prevent them from interfering with fighters on Combat Air Patrol (CAP). I.e. by using a 50nm range limitation. Or preventing use of the new 160nm-range SM-6 SAM against difficult targets at ranges beyond 50nm.

Self Defence

Normally, only one friendly unit engages an enemy contact at a time. However, in many cases it would make sense for units to defend themselves when they feel threatened, even when other units are currently firing at the contact. For example, a pair of MiG-23s are closing on a bogey at nearly 2000 knots. Seconds before reaching firing parameters for an AA-7 Apex shot, a SA-5 Gammon site launches a salvo. If the MiG-23s wait for the SA-5s to reach the target before shooting, they will already have merged with the contact and the dogfight begun. To prevent this from happening, it is possible to configure the Self-Defence Range in the WRA. Typically, the self-defence range for the AA-7 is 5nm. This means any MiG-23 that gets within 5nm will launch AA-7s regardless of how many weapons have been allocated to this bogey by other units.

Another example would be a Carrier Battle Group (CVBG) under attack. The escorts do a great job picking off incoming anti-ship missiles with SM-2 SAMs. But once a leaker gets within 5nm of the carrier, the ship will fire a salvo of Sea Sparrows or RAM in self-defence regardless of whether the anti-ship missile currently has SAMs from any of the escorts targeted at it.

Conclusion

In order to control automatic use of weapons in Command: Modern Air / Naval Operations, first set the 'Weapon Control Status' (WCS) in the Doctrine window to FREE / TIGHT / HOLD. When set to FREE or TIGHT, weapon employment can be further refined by target types and the number of weapons to be fired in the 'Weapons per Salvo' column in the Weapon Release Authorization (WRA) tab. The firing range can be limited in the 'Automatic Firing Range' column and the AI can also be prevented from firing the weapon in this column. Finally, the 'Self Defence' column allows a unit to fire the weapon regardless of whether the target being dealt with by another unit.

Please note that the decision of whether or not to shoot, and the number of weapons to fire, is decided the instance the simulator generates the weapon salvo. It uses what is currently known about the target to determine what target type settings to use. Two incoming anti-ship missiles may be of 'Unknown Type' and 'Supersonic Sea Skimming' respectively, and may trigger very different behaviour. So make sure that the target types are configured correctly for all target identification levels, from Unknown and Unspecified target types to Specified target type.

It should be noted that guns are normally set up to automatically fire at anything that comes within range. Default settings allows the weapon to be used against any legal target type, using continuous fire, an unlimited number of firing units to participate per target, employment out to max range, and self-defence out to max range. This means that guns will always fire even when the target is under attack by other weapons. To limit

gun usage, make sure to alter the 'Self Defence' range in addition to any other custom modifications. Nerf Wars: On downgrading Russian systems & units in Command

Nerf Wars: On downgrading Russian systems & units in Command

So, ever since it became clear that Russian combat performance in Ukraine has been “less than stellar”, there has been a persistent request in many corners of the web towards the Command dev team. The request can be summed up as:

“So, when are you guys going to nerf Russian equipment in Command’s database, to match what we are seeing in Ukraine? It seems it is performing well below official specs.”

(Oxford Dictionary: “to nerf”: (of a video game developer) reduce the power of (a character, weapon, etc.) in a new instalment or update of a video game.)

A naval-oriented variant of this comment is: “I tried simulating the attack on the cruiser Moskva, and I just couldn’t sink it as happened in real life. So CMO is probably talking up Russian hardware.”

Now, there is a short and direct answer to both these claims, but we have been advised not to print it here. So let’s go into the more elaborate and slightly more polite version instead.

Command, by default (ie. stock DB values) represents Russian systems (and all other nation’s systems) as they are meant to be used, by trained crews employing them according to their design doctrine. Shortly after the Ukraine conflict got going for real, a US general remarked that “Russian hardware works pretty well... when used by Ukrainians”.

(As a quick example of this, Carlo Kopp had written an unusually insightful analysis on what happens when Russian SAMs get used correctly, and also what happens when they are not. Favorite quote: “The Syrians used mobile missiles in a fixed configuration; they put the radars in the valley instead of the hills because they didn’t want to dig latrines — seriously.”)

At the same time, Command also recognizes that combat is not a sterile hardware contest, and provides a lot of “soft factor” options (crew proficiency settings, reaction times, doctrine settings, EMCON, ROE etc.) to offer the ability to represent sub-optimal usage of said hardware. You can have two different units use exactly the same hardware with vastly different effectiveness and survivability (our favorite reference example is Iraqi 1991 SA-6 battery vs. Serbian 1999 SA-6 battery).

An example list of the factors you can edit:

Proficiency levels, both side-wide and per-unit

Reaction times (OODA loop values)

Doctrine settings

Rules of Engagement

EMCON settings

WRA settings (ie. firing doctrine)

Per-component equipment failure

That very last aspect is particularly important in light of what has been recently learned WRT the system readiness on the Moskva. According to the leaked readiness report (standard caveat for CYA-shaped leaks), both the central SA-N-6 fire control radar and the short-range SA-N-4 systems were effectively disabled due to equipment malfunction and were waiting for replacements. In addition both the main guns as well as the main air-search radar were also inoperative. In other words, the Moskva was sent to an active warzone almost naked against air/missile attack.

Can this be faithfully reproduced in Command? Absolutely.

Can other factors, such as apparently poor crew proficiency which led to both poor reaction times and also abysmal damage control, also be modelled? Also yes, very easily so.

So why then do some CMO users out there find it so hard to reproduce results like the Moskva sinking? (Or the Saki airbase strike, or S-300/400 batteries not being omnipotent, or UAVs apparently roaming at will, or...)

TOUGH LOVE WARNING: Because some people's idea of "testing something in the scenario editor" is to quickly plop down a few units here and there, in their stock-DB setups, give them free fire reign against each other, and call it a day. No customization for soft factors, component status or any of the other real-world aspects that directly impact the end result.

Thankfully, some players are disciplined enough to do it right. Here is an example, soon after the actual sinking itself when info was still scarce. (This of course is far from "the last word" on the subject. SMEs are still debating e.g. how the Sheffield went down after a single Exocet hit while the Stark survived two of them. Hell, even WW2 events are still up for analysis. It's a safe bet that the Moskva sinking will be endlessly discussed by our grandchildren.)

It is tempting indeed, to embed the soft-factor issues directly into the DB entries themselves, so that when you spawn a Russian unit on the virtual battlefield, it's in a sorry condition out-of-the-box, ready to be clobbered. The price you pay for this expedient approach becomes obvious only later, when you realize that not only you lost the ability to clearly separate man (and context/circumstances) from machine, but you also railroaded yourself intellectually into believing that this Russian unit will always behave like this.

Does this matter? Let us consider, for instance, an Israeli staff sergeant nerfing Egyptian tanks in a wargame just prior to Yom Kippur in 1973 "because they were such pushovers just a few years back". Did this specific example happen? Maybe, maybe not; but we know for a fact that the Israeli military establishment grossly underestimated Egyptian & Syrian forces because of their lightning successes in 1967 (they essentially nerfed them in their mental "databases"), and that plenty of Israeli soldiers paid for that intellectual myopia with their lives. Is this a mistake we want (or can afford) to repeat?

Such a radical shift in combat effectiveness with identical/similar hardware does not happen just between conflicts, but also within the span of a conflict itself. Returning to Ukraine for an example, in the early days of the "rush for Kiev" we observed a lot of Russian SHORADS batteries getting bombed by aircraft while completely inoperative. As it turns out, apparently the rapid speed at which these elements were forced to move (to screen the assault forces) prevented them from properly screening/leap-frogging each other and thus actually operating as they are designed and supposed to do. When later these very same systems were properly echeloned with the forces they covered during the Russian withdrawal to Donbass (and also undeniably as the hard lessons of the first weeks were distilled to the surviving operators), their effectiveness

and survivability were restored to expected levels. (And then the Ukrainians introduced HARMs in the theater... but that's another story)

How can you represent such drastic differences in effectiveness in the very same unit, if the soft-factors are embedded in the database entry? Short answer: You can't.

(Longer answer: You can cheat/hack your way into it by having multiple entries in the DB, each representing different competence levels and equipment maintenance. It's a very hacky solution, a maintenance nightmare, and again you are mixing up man, machine and context. We sometimes were forced to do something like this back in the computer-Harpoon days, simply because Harpoon had absolutely no soft factors. Nowadays we can and must do better.)

So, to recap: We would be doing a grave disservice to both our commercial players and especially pro customers by directly embedding soft factors into the DB just so that Joe Player can get a realistically-degraded Moskva out of the box. Platforms and systems in the DB are spawned in pristine condition and (by default) are assumed to be crewed competently: This is not a design oversight, but a conscious and carefully-considered decision. Players can then modify the platforms themselves, turning them into anything from decrepit spank-targets all the way to invincible fortresses, and shape the context of the virtual environment in their favor or against them, in order to either recreate historical situations or explore hypotheticals of past, present or future. But in every case, it's something that they will have to roll up their sleeves and do themselves. To quote Norm Koger from two decades ago: "Word processors ask a lot of those who would use them to create stories".

The v1.03 Command update

Major improvements to sonar model. Based on an ongoing discussion of existing flaws in the sonar model (see here for the tech background) with various SMEs both on and off the forum, we have renovated many aspects of the sonar model in order to improve its real-world fidelity. A brief summary of the changes included:

- The thermocline layer is generally weaker then before for a given location, and completely stops being effective at 70+ degrees north or south latitude.
- The minimum bottom depth necessary to achieve converge zone (CZ) propagation is sharply increased per given location, and is no longer static but dependent on latitude. CZs are now also impossible at 70+ degrees north or south latitude.

– Most under-layer sonars (SOSUS/SURTASS excluded) lose the 2x detection range bonus afforded by the deep sound channel. This sharply reduces the in-DSC detection ranges of VDS & towed arrays.

Combined, these changes have the effect of making submarines at great depth more difficult to detect with towed arrays or VDS, and shifting the emphasis more to direct-path detections at the expense of CZs.

Select IRST/FLIR systems can now produce AAW fire control-grade data, allowing EMCON-silent engagements with AMRAAM-class weapons (background discussion here). Note that the fire-control data generated by advanced IRSTs can be used not only for direct onboard shooting (e.g. F-18E uses IRST-21 to silently shoot AMRAAMs at target) but it can also be distributed to other platforms via CEC (e.g. F-18E silently tracks target, hands off to CEC-equipped ship which silently shoots SM-6 at target).

NOTE: Scenarios will need to be made/rebuilt in DB3000 v488+ in order to use this feature. These are the sensors in DB3000 v488 that have this ability.

Visual and IR sensors are now also susceptible to look-down clutter. For example it is easier for an IRST (or the plain Mk1 Eyeball) to pick out an aircraft over the horizon line than against the surface background. This provides an extra incentive to "go low", even for VLO aircraft, and makes terrain-hugging cruise missiles even harder to spot.

Datalink + TARH AAW missiles (e.g. AMRAAM, SM-6 etc.) can now go active without immediately going autonomous. This allows such missiles to be guided with greater precision & reliability against VLO targets or in strong OECM environment. The datalink is cut instead only when the missile acquires the target (or impacts on it, still under positive control from the link provider).

Various improvements in ballistic missile & BMD modeling, including fixing the simulation slowdown caused by ABM DLZ calculations.

Numerous smaller changes and quality-of-life additions. As a quick example, it is now much easier to visually spot when a side/group/unit's doctrine & ROE settings are overridden at a higher (parent) or lower (children) level, as they are now displayed in a different color when they do. Also as a popular request, the LOS-tool now supports altitude input in feet in addition to meters.

The v488 release of the DB3000 database, incorporating countless player requests and new additions as well as numerous tweaks and fixes. You can view the complete DB changelog [here](#).

Command: Modern Operations – The new simulation & editor features

<https://command.matrixgames.com/?p=4990>

We have already covered the massive UI/UX changes and improvements in CMO in a previous two – parter piece. We also saw the significant improvements in ground operations. But at the end of the day, Command's beating heart remains an elaborate simulation engine of air, naval, strategic and joint operations at the tactical and operational/theater level, together with a powerful editor for manipulating and creating with it. So let us take a look at some of the major new features in this area.

Turbo-charged for your pleasure

One of the most immediately noticeable features, especially when running large or elaborate scenarios, is the jump in performance. Command now offers both greater absolute speed in simulation execution (ie. reduced "pulse time" as displayed when the "Show Diagnostics" option is enabled) as well as significantly increased CPU utilization, as a result of improved parallelism. (As an example of the latter, on a recent trial run of "Northern Fury #9: Hold the line", CPU load was observed fluctuating between 65%-75% on a Core i7-6700 system. This may vary across different systems, of course). Maximum practical unit count has also increased thanks to better memory management, with the result that even more massive scenarios are now feasible.

Until the COWs come home: Comms disruption, aircraft damage & new weapons

As we previously mentioned, it was decided to merge the previously DLC-locked features of "Chains Of War" to the core simulation available with Command. This allow both scenario authors and players to make extensive use of these features without any licensing concerns. We've covered these features extensively before, and we included cargo, amphibious and airdrop operations on our previous article on ground warfare, so here is a short summary of the others:

Comms disruption explores the different & numerous ways that a unit can be isolated from its parent side's communications grid, and the repercussions of "going offline" (spoiler: none of them are pleasant). This feature

has been used very successfully on COW, Shifting Sands and The Silent Service, and now it can be used on any scenario. One of the post-launch improvements of this feature is that when a unit re-joins the comms grid, the UI provides much more finegrained feedback on what type of information was updated for a contact (if any), and skips mentioning contact merges that produce no new info. Ready to feel a cold sweat as your units abruptly go out of visibility and out of your control?

Detailed aircraft damage is arguably the most popular feature of the bunch, enabling a much more finegrained level of damage & attrition on aircraft than classic CMANO's "kill on first hit" paradigm. Since its original introduction, this feature has been significantly enriched with feedback from both commercial & professional users.

Advanced weapon types: High-energy lasers, railguns & HVPs, and tactical EMPs are now available at your disposal, to mindlessly include in the next Transformers flick use in all scenarios addressing the challenges of warfare in the foreseeable future.

Alone and unafraid: Realistic submarine comms

One of the benefits of absorbing the COW features is that now we can freely use them as building blocks for additional functionality, inserting features & mechanics that would previously require some non-trivial Lua elbow grease. Realistic sub comms are one such example.

When this feature is enabled, submarines who go below 40m depth go off the communications grid. As in other cases of comms disruption, they are no longer under direct player control, and only their last reported location is available (this BTW means that sending them deep without first tasking them with a mission may be pointless, since they will simply sit there). The player can, at any time, send a "bell-ringer" ELF signal to a no-comms submarine to recall it to re-establish comms (right-click on sub icon, click on "Summon to re-establish comms"). The sub will attempt to rise to shallow depth to rejoin (this may not be immediate, as it will try to evade nearby hostiles). Once it rejoins, the sub will share its contact updates with its parent side.

Never stray from the path

One of the more persistent problem of CMANO's navigation AI was that its pathfinder was binary-based: Either a spot was passable by a unit, or it was not. This resulted in a tendency in units to aggressively cut corners in their plotted paths in order to travel to their destination as efficiently as possible, sometimes coming dangerously close to shore or other obstacles. This can result in some unintended "beach surfing", as this example demonstrates:

In this case, the Freedom is placed in the mouth of the Hormuz straits and ordered to travel inside the Persian Gulf. It plots an efficient path — too efficient, as it nearly clips the shoreline at Khasab. There is a possibility that the ship may get stuck on the shore, if left unattended. Ships generally don't navigate this way if they can help it.

To address this issue, we implemented a new cost-based pathfinder that evaluates a location's suitability based on a number of different factors. In the case of ships, a primary concern is local depth and proximity to terrain; generally large ships tend to prefer to maximize both, while smaller ships are somewhat more

freewheeling. Using this new logic, ships are now able to plot much more realistic (and gameplay-friendly) courses, as the very same example played out in CMO demonstrates:

Stay in formation

This has been a popular request, which we are finally happy to oblige: The formation editor can now also be used to define and edit aircraft formations. It sounds like a simple change, but in reality we had to go through extensive reviews of the mission AI logics to make sure that this did not interfere with aircraft assembly & grouping logics (in some places it did, and we had to make adjustments). But we think it's worth it, and you will probably agree: This opens up new opportunities for formation tactics and optimum placement & distribution of air assets in a group, both for regular transit or surveillance as well as combat.

A patrol of four MiG-31B sweeps over the sea north of Murmansk. The group is in loose line abreast formation, to widen the search area. Lead and the nearest wingman are already in position, while the two outer members race to catch up to their stations.

I like the way you move (and think)

Various tweaks & improvements have been applied to the kinematics models & AI routines, with a strong emphasis on aircraft and missiles. Among other items:

Numerous tweaks to aircraft flight model, specifically for “combat” conditions. For example, aircraft no longer “wobble” between headings as they must first roll towards the turn direction before committing to a turn. This in turn makes inherent roll-rate more important to close air combat maneuvers (This is easier to observe in the Tacview window).

Helicopter diving rates have been reduced (they were previously dropping like a stone)

Surface- and underwater-launched missiles now use the same improved pitch kinematics as air-launched missiles, resulting in more realistic trajectories.

Lofted AAW missiles now begin their terminal dive earlier, to avoid a too-steep approach to the target.

Air combat AI improvement: Aircraft now consider approaching fighters/interceptors as imminent threat, not just missiles. This helps AI-controlled aircraft perform more proactive evasive maneuvers against fighters about to perform gun attacks on them (e.g. MiG-17 vs F-105).

Significant change in unit AI logic: The “evaluate targets” and “evaluate threats” logics are now not performed on every sim pulse, but instead on regular intervals dictated by the OODA-Targeting (modified by crew proficiency) and OODA-Evasion values respectively. Therefore these two OODA values, combined with crew proficiency, become even more critical to a unit's effectiveness and survivability.

Feeling the strain: Aircrew G-tolerance

If you have ever closely watched a dogfight in CMANO, one of the things that may seem odd if you are familiar with real-world ACM is the seemingly iron-man constitution of the virtual fighter pilots; they seem to be constantly engaging in turns, climbs and dives as tight as their airframes will allow them, without regard for their own fatigue from the continuous high-G acceleration loads. This isn't how close air combat works in real life; even the best pilots need a breather between aggressive maneuvering in order to allow their bodies to recover, otherwise they become more dangerous to themselves and their side than to the enemy (yes UCAV fanboys, you can laugh now). So we set out to model this nuance in CMO.

When an aircraft is considered to be performing "combat maneuvers", an extra UI element is shown on unit status panel:

This represents the crew's tolerance to hard maneuvering. The longer the aircraft is continuously pulling a hard turn, the more this buffer fills up. (Getting out of a hard maneuver, e.g. reversing a turn, reduces this strain and allows recovering. This is one of the reasons that scissors are a popular practice in RL dogfights.)

Once the tolerance is exhausted, the crew begin suffering G-LOC and have to significantly relax the turn (this is easier to notice in Tacview, as the aircraft wing bank visibly changes, but you should also observe a noticeably larger turn radius on the map's top-down view as well), while regaining their senses and control of the aircraft. Of course the aircraft is particularly more vulnerable during this recovery period, which makes cooperative tactics and mutual support all the more important. The maximum tolerance level depends on aircrew proficiency, so experienced pilots get one more significant advantage against greener adversaries (Ed: Like they need one more...).

Radar frequency agility

Not all radars are created equal. One particular feature that became quite important during the Cold War for radar performance against jamming and other countermeasures was its ability to "frequency hop", ie. shift its operating frequency to a different value within its feasible operating range. CMO now models this, and two particular operational benefits of this ability:

Frequency-agile radars are significantly harder to noise-jam, as they can shift to a new frequency once their existing one gets flooded with static. The jammer has to either start hunting for the new frequency, effectively hopping after the radar operator, or alternatively flood the entire frequency range with jamming power to make hopping pointless; both counter-actions are feasible but they both have complications of their own.

These radars are significantly less susceptible to doppler-notching maneuvers (more on this below).

This feature became so important to radar engineering & operations from the mid-1960s that, quite often, the main difference between a very capable domestic radar set and its export-cleared variant was the physical removal of the frequency-hopping circuits in the latter kit.

As in other cases, electronic-scan arrays and especially AESAs get massive advantages; in this case, all P/AESAs are automatically considered as being frequency-agile, while older mechanical-scan sets have to explicitly have this declared in the simulation databases.

A notch on the bedpost

Doppler notching, a maneuver designed to exploit a fundamental flaw in pulse-doppler radars in look-down mode, has been included until now abstractly in CMANO as part of the weapon endgame calculations. This time, however, it can be actively and preemptively used as a maneuver, both for missile track lock-break and for general surveillance radar detection avoidance.

Aircraft can attempt to fly perpendicular to an emitter using doppler filtering in order to hide inside its “blind” velocity gate. The effectiveness of the maneuver varies with crew skill (an “ace” pilot will execute it far more effectively than a novice), to discourage manual micromanagement by players. Aircraft under missile attack with a doppler radar guiding the missile will also actively try to beam the radar instead of the missile (the geometry of the two axes can vary significantly). The maneuver is ineffective against pulse-only radars and less effective against frequency-agile radars.

Players can also deliberately plot courses for aircraft that fly perpendicular to known PD search radars, to reduce the actual detection range. (If you’re a Microprose F-19/F-117 virtual vet this may bring back some memories).

Patrolling in style

This has been another popular request: “I like the flexibility of the patrol mission, but sometimes I want to enforce my custom patrol movement patterns instead of the ‘random within defined area’ logic, even at the cost of predictability”. Well ask, and ye shall receive – in the form of an extra patrol mission option: movement style.

This option has two settings: “Random within area” (default) and “Repetable loop”. “Random within area” is the patrol behavior you are already familiar with. If “Repetable loop” is selected instead, the assigned units treat the reference points of the patrol area as a racetrack, just like in a support mission. This allows the player (or scenario author) to define precise, consistent patrol patterns (figure-8, expanding spiral, ladder etc.) while also retaining the “investigate & prosecute” behavior that distinguishes patrol missions from support ones.

When the “Repetable loop” setting is active, the patrol area is depicted very similarly to support mission racetracks, to emphasize the difference:

Till death do us part: Merge scenarios

This is another gift from the PE world, and we bet it will be a much-appreciated one. If you are a scenario author, you are already familiar with the base & group import/export (ie. the .inst files), and how convenient and time-saving they can be when putting together the elements of a scenario. This “construct once, re-use everywhere” concept has now been taken a step further: You can make individual scenarios and then merge them into a single scenario.

Scenario #1 is treated as the “base” for the merge, so behind the scenes you are essentially cherry-picking all transferable items from scen #2 and adding them to scen #1 (the log output also makes this clear). Elements that are “atomic” or otherwise hard to merge (date & time, weather, title & description etc.) will, on the output, be those of scenario #1.

The power & versatility of this tool should be readily apparent to anyone who’s delved in scenario editing. Now you can create once, store and re-use not just groups or bases, but multiple foundational scenario elements: Sides, missions, events, actions, conditions etc. We have been repeatedly amused by the wacky, quite unforeseen ways the original import/export functionality has been employed by industrious scenario creators, so naturally we are curious to observe how they’ll bend this one to their will.

Lua loco

The Lua API continues its expansion in CMO and offers additional hooks into the simulation engine as well as various methods for pulling the strings of the running scenario. One of the new hooks ties directly into the AI model: You can individually instruct units to, quite literally, not think for themselves (You in the back, quipping “you mean they do this now?” – SIT DOWN!). More specifically, you can set individual units to skip their AI routines for evaluating valid targets and picking out the primary one among them. This has two direct benefits:

It makes it easier to implement custom targeting AI routines in Lua, since an author not longer has to “compete” with the build-in AI for this.

As these routines are among the most CPU-expensive pieces of the simulation pipeline, disabling them can have a drastic effect on the speed & scalability of a large scenario. For example, if you disable the AI cycles of all static/inactive buildings, then only “active” units will use the CPU for this work (internally Command already does a lot of such optimizations, but since it cannot “intrinsically” know which units are static & inactive, it has to check them, which itself is not free).

Additionally, this ability can allow simulating “dormant” states for units (e.g. units begin a scenario in a “comatose” state, but later because of XYZ they become activated).

12.0 Glossary

AAA: Antiaircraft Artillery

AAM: Anti Air Missile

AAW: Anti Air Warfare

ASuW: Anti Surface Warfare

ASW: Antisubmarine Warfare

BINGO/ BINGO FUEL: Typically for aircraft. Bingo is the moment or condition that the aircraft has only enough fuel to make it back to its own base at a normal cruising speed and altitude.

BOL: Bearing Only Launch. BOL capable weapons may be launched down a bearing toward a target. Most BOL weapons have an activation point where their internal sensors are enabled. The activation point is the point selected when ordering the launch. With the launching unit selected, use Keyboard Command for BOL launch is CTRL + F1, and select the activation point on the map.

DLZ: Dynamic Launch Zone.

ECM: Electronic Counter Measures

GMT: Greenwich Mean Time.

HVU: High value unit.

LOS: Line of Sight

NEZ: No escape zone.

OOB: Order of Battle, a detailed list of units and equipment in a side's inventory.

OECM: Offensive Electronic Counter Measures.

ROE: Rules of Engagement.

RTB: Return to Base.

Winchester: Empty. Game evaluates against a role value for this in determining whether to order an aircraft to return to base.

13.0 COMMAND Update History

COMMAND has changed considerably from earlier versions. Many new features have been added and many UI changes have been made. While much of what applied to early release versions still applies to the latest version, much else does not. As many returning players have left before a specific update time, cataloging all the major changes in earlier releases is in order.

For players whose first introduction to COMMAND is the current “Command: Modern Operations” version, much of this will not be of any concern. But for players who have been used to the very different mechanics of earlier versions, reading these updates may be helpful to avoid feeling out of place.

Version 1.01 (October 2013)

=====

This was a “quick-fix” update of glaring issues found right after release. Notable updates include:

A dedicated “weapons” page in the database viewer.

The ability to deselect all reference points via either a menu or by pressing CTRL-END

-The ability to zoom to the mouse location instead of the camera center, for the convenience of say, focusing on a contact on the far side of the map.

-A new game option: Sonobouy visibility. The choices are Normal (sonobouys are prominently displayed), Ghosted (they appear, but in a “faded”, translucent way), and Do Not Show (not visibly appearing on the map at all)

-A new dedicated “weapon” page on the Database Viewer. The characteristics of weapons from unguided bombs like the Mk82 to super-smart munitions like the LRASM appear. The weapon’s DB entry includes its range, maximum speed/altitude, warhead size, signature, and seeker type/ability.

-An “Abort Launch” button in the air ops window. When pressed, any selected aircraft beginning the launch sequence will stop and attempt to return to an open parking facility.

-Hold-fire can be issued to multiple selected units.

-The “Add Unit” window now remembers the filters used in the last unit addition whenever it’s reapplied. For instance, if the “Ship” type filter is selected and the word “Iowa” is typed into the “class” text box before the window is closed, reopening it will have a filtered list of Iowa-class battleships in plain view.

-Major improvements in aircraft dropping ordnance without manual control.

Version 1.02 (December 2013)

=====

This was the first major update. Major changes included different realism settings (see 5.4.9, Scenario Features + Settings) and the first side proficiency changes (see 5.4.4, Scenario Options, Proficiency Settings)

-Per-waypoint speed, altitude and sensor commands. Select the waypoint and either right click or press F2 or F9. The unit will obey the commands when it reaches the waypoint.

-Quick Jump slots.

Quick Jumps can be used to rapidly shift back to a unit or contact. To use quick jumps:

- Select a unit/contact and zoom the camera to the desired level.
 - Press CTRL+[NUM] (the number can be 1, 2, 3, 4, etc...) and the camera altitude and location will be saved
 - To jump to that unit/location, either press the initial number or use the "quick jump" top level menu and select from the list. The saved numeral is to the right of the unit in the menu for easy reference.
 - Note that quick jumping doesn't actually select the unit/contact, instead it merely moves to its location.
- New game option: Reference point visibility. The options are "normal", "small", and "do not show", the latter two serving to reduce clutter.
- The ability to set the activation time in the mission editor. This can be vital for both hands-off normal play and for scenario creators wishing to time missions launched by computer controlled sides.
- Side-level proficiency features, ranging from Novice to Ace. Aircraft turn rates and weapon evasion calcs are affected by side proficiency, as is the reaction time of all units.
- The map cursor databox (aka Black Box of Data) can be set to appear next to the mouse cursor, at the bottom of the map, or not show at all.
- New feature: Mouse preview mode. Hovering over a contact now displays the selection and datablock information for that unit. It's useful for checking contacts quickly.
- Can now select new home bases for multiple-selected aircraft.
- New title screen selection: Resume from last autosave. This, true to name, loads the last successful autosave. This makes returning to play after an issue easier and more convenient.
- Background music is now available. While only one track is included with the game, more can be manually added.
- New Realism Option: Unlimited Aircraft Weapons. When checked, this allows aircraft to have unlimited amounts of any loadout available. While this can be unrealistic (ie, compare the shortage of air-launched Exocets in the Falklands War to the option of having all the Etendards launch repeated mega-salvos with this checked) and unbalancing (the scenario may be predicated on the player having small numbers of certain weapons at the air base), it can be useful for quick scenario editor experiments or scenarios where neither is an issue (ie, the available aircraft can only carry basic unguided weapons, but they would plausibly be present in large numbers)
- Two significant changes to "bingo" (unable to safely return to base) fuel calculations. First is different margins. "High value" aircraft like tankers and AEW planes will play it "safer" and automatically return on larger amounts of fuel than "expendable" aircraft like fighters. The second is that any aircraft with fewer than 1,500 liters of fuel is considered "bingo" no matter how close or far from an airbase it is.
- "Strategic" (ie, ballistic missile) submarines now automatically move to avoid any potential surface or sub threats in their proximity. They're there to launch missiles, not engage in torpedo fights.
- New Doctrine Option: Use SAMs in anti-surface mode. This enables or disables the use of naval anti-air missiles in an anti-surface mode. There are valid conditions for using SAMs freely against surface targets and valid conditions for saving them for aircraft.
- New Patrol Type: Sea Control Patrol. This folds ASuW [anti surface] and ASW [anti-submarine] patrols into one. Units under a sea control mission will engage both surface and underwater targets.

-Manual deployment of dipping sonar. Pressing shift+D with an appropriate helicopter selected will have it hover low over the water and deploy its dipping sonar.

-Can now edit names and delete specific aircraft directly through the air operations window.

-Sonar and sonobuoy mechanics modifications. First, older and/or lower-frequency sonars are affected more by reverberation in areas like near ice packs or shallow water. Second, aircraft are now constrained by datalink limitations when deploying sonobuoys.

-Secondary windows now remember their last positions and sizes between sessions of running COMMAND. This helps users keep to a familiar pattern.

-New doctrine option: Engage targets of opportunity. This means that units will engage any contact with which they have suitable weaponry for even if the mission tasking doesn't specifically call for it.

EXAMPLE: A carrier group, including escorts with anti-ship missiles, is set to run a routine support mission. If "engage targets of opportunity" is engaged, the escorts will automatically fire their ASMs at any appropriate contact when in the proper range. If it is not engaged, they will not unless the mission is changed or they are manually ordered to do so.

-New event trigger: Unit Detected. The event will trigger when a unit (filterable to varying degrees) is detected by the specified side.

-New event editor change: Event conditions. If included, the event will only fire "IF" certain conditions, in this case, side posture, are met. (IE, IF the "USSR" side is hostile, the NATO side will fire its nuclear missiles at 0430.

-"Hold Position" command added to the right-click menu for mobile ground units. This can be used to keep them in place if they've moved an inappropriate distance.

-Database hyperlinks added for aircraft (in the "air ops" window) and weapons (in the "unit weapons" window). Clicking them will bring up the DB viewer with the appropriate entry.

Version 1.03 (March 2014)

=====

This update was focused on performance, stability, and database issues.

-Aircraft loadouts now affect drag (all those bombs hung underneath that F-105 are not going to help its agility).

-A staggered list of autosaves, where the most recent autosave is "autoscen_20", with each autosave pushed back 20 (real world) seconds until autoscen_100 is reached. This can help with finding issues, as an autosave shortly *before* (instead of after) the issue can be reached.

-Patrol throttle option added. Units with no other override obey this setting. It can be used if the goal is persistence (creep/loiter), covering more ground (cruise) or moving very quickly for the sake of evasiveness (full/flank).

-Aircraft with an attack profile lower than the minimum release altitude of their carried weapons will try to fly just above the latter so that they can reliably and precisely drop their weapons (even if it's at the expense of being more vulnerable). Previously, they dropped at maximum altitude, which was poor for any kind of unguided accuracy.

-Floating mines can now be detected by visual sensors (including crew Mk1 Eyeballs) on ships and low, slow (less than 150m/450 feet and slower than 200 knots) aircraft.

-CTRL+Left Click and Move will create a new waypoint.

-Major new game option-no pulse mode. The map and UI refresh every 0.1 seconds and the engine attempts to execute each turn as rapidly as possible. The result is a considerably smoother, quicker-running experience. (No-pulse is the default mode in 2.0)

-Engine throttle now affects the IR signature of an aircraft.

-Whenever an aircraft being prepared is missing primary stores (for instance, if an airbase has one sortie's worth of bombs, they're used up on the mission, and the survivors attempt to rearm for another go-round), the outfitting is cancelled and the aircraft reverts to "reserve-clean" loadout.

-Selecting a unit now highlights and "thickens" its weapon range symbols in addition to the sensor symbols.

-Aircraft returning to base for reasons other than low fuel now return at full rather than cruise throttle if possible to reduce the time of vulnerability.

-Unknown contacts can now be designated as friendly with F and neutral with N.

-Improved weather effects modelling. Progressively thicker clouds have a progressively worse effect on EO and laser sensors (including laser designators).

-Significant tweaks to "air boss" AI to unjam runways, maintain formation cohesion and keep from needlessly sending aircraft back to their hangars.

Version 1.04 (July 2014)

=====

This update introduced a strike "auto-planner" and other improvements such as prosecution areas for patrols (see 7.2.3, Patrol Missions) and sprint-and-drift movement for warships.

-New option to turn off the BMNG layer, which returns the map to a "retro" wireframe format.

-Added quick jump top level menu which lists all saved quick jump slots.

-Prosecution areas (additional zones that patrolling units will not enter normally, but will pursue contacts in) added as an option to patrol missions. Prosecution areas act as a way to restrain patrolling units. For example, with a prosecution area, a patrolling aircraft neither breaks away from its scheduled patrol to investigate a contact far away from the combat area nor tries to follow retreating enemy fighters all the way back to airfields with known SAM batteries around them.

-Added double-click functionality.

- Plotting course (F3): double-click when placing the last waypoint when plotting a course to exit Plot Course mode.

- Using Distance/Bearing tool (Ctrl + D): double-click to exit Distance/Bearing tool.

-Added CTRL+click and drag functionality. By pressing the CTRL key and clicking-dragging a waypoint or reference point that already exists, a new one can be made.

-Most hotkeys now have a menu item. Three new menu categories added ('View', 'Contact' and 'Quick Jump')

-Added new window accessible via "Help-Hotkeys" that displays all hotkeys.

Added new hotkeys, including:

V - Switch unit/group view, for laptops without a numeric keypad

Ctrl+V - Toggle God's Eye view

F11 - open Mission Editor

Ctrl+F11 - create new mission
Ctrl+F9 - unit/group Doctrine + EMCON
Ctrl+R - rename Reference Point (not sharing 'R' hotkey with Rename Unit)
Ctrl+Del - delete selected ref point(s)
Ctrl+Ins - add ref point
B - return to base
Ctrl+M - clear message log
A - Toggle Hold Fire for selected units (no AI attacks)
Ctrl+A - Toggle Hold Fire for all units on current side (no AI attacks)
I - Toggle Ignore plotted path when attacking for selected unit
Ctrl + I - Toggle Ignore plotted path when attacking for all units on current side
E - Drop Target(s)
Ctrl + E - Disengage (Drop all targets)
L - Hold Position for selected unit(s)
Ctrl + L - Hold Position for all units
Alt+S - Toggle sides (switch sides) in scenario editor
P - Drop contact(s)

-Ability to configure ghosted unit visibility for groups under “map settings-show ghosted group members for...”

-Ability to configure plotted course visibility under “map settings-plotted courses”.

-For ease of examining the mission details, the assigned mission description in the unit status window is now a hyperlink. Clicking it opens the mission editor with the mission in question selected.

-Detected non-friendly unit groups (ie, airfields, naval bases, mobile groups) are shown as groups (one “group” icon) rather than as units (many individual icons) when in “group view”.

-There are now eleven preset cloud cover profiles, with the mouse pointer containing more information on them:

- Thick fog 0-2k ft, solid cloud cover 7-36k ft
- Thin fog 0-2k ft, solid cloud cover 7-36k ft
- Solid middle clouds 7-16k ft, moderate high clouds 30-36k ft
- Moderate middle clouds 7-16k ft, light high clouds 27-30k ft
- Moderate high clouds 25-28k ft
- Moderate middle clouds 7-16k ft
- Moderate low clouds 2-7k ft
- Light high clouds 20-23k ft
- Light middle clouds 10-16k ft
- Light low clouds 5-7k ft
- Clear sky

-Offboard sonar sensors (towed arrays, VDS) get half the own-powerplant noise reductions as hull sonars and a further reduction from the existing thermal layer and deep channel modifiers-there is a reason units that have them are considered highly capable.

-When the player attempts to load a scenario that uses a missing database, COMMAND informs them of this and aborts the load.

-The AI can now fire heavyweight torpedoes (ie 650mm Soviet monsters) at a maximum range of 8 nautical rather than the minimum of six kilometers.Later updates would allow for more fine-tuning of torpedo launches.

-The database “Weapon” page now features guidance details for the currently displayed weapon, useful for, say, seeing what versions of a missile have an IR seeker and which have an SARH seeker (the Soviets loved making dual versions of the same airframe).

-The AI can now engage enemy SAMs in anti-surface mode if able.

-Significant improvements to intercept logic against extremely fast targets (such as ballistic missiles and satellites). Launcher units begin tracking at much longer distances and ensure all mandatory datalinks can be established.

Version 1.05 (September 2014)

=====

This update was the first to be released on Steam and introduced a huge reform of aerial refueling operations, non-cooperative target recognition radars, and “cranking” (slowing down in aerial combat as much as possible to reduce the time to close the distance).

Aerial Refueling: This update introduced many changes to help aerial refueling. The three biggest changes were:

- Crews becoming better and more context-aware about picking tankers.
- Tankers becoming better with managing their own fuel, to prevent aircraft completely emptying them.
- Aircraft returning from missions no longer headed for tankers if not necessary.

There were subsequent tanker improvements between this and V2.

NCTR: Certain radars have a “non-cooperative target recognition” function. As of V2, the database entry for a unit will, on its sensors list, contain a note if the radar has NCTR capability or not. The two types of NCTR radars are:

JEM-Jet Engine Modulation. This requires a “look” at the blades of the engine, and thus requires the target’s front to be facing the radar (within 15 degrees).

NBILST-Narrow Beam Interleaved Search and Track. This is more advanced and can work from any angle.

BVR Cranking:

Starting in this update, aircraft launching BVR missiles proceeded to “crank” after firing. Cranking means the launching aircraft will simultaneously slow down and turn as far away as possible from the target while still being able to guide the missile. The goal is to preserve the range advantage as much as possible. The degree to which aircraft can crank is now customizable in V2.

Damage Reform

-Damage rates to vehicle subsystems increased, messages updating the player on their status appeared in the message log, and the damage control window now color-coded subsystems in various levels of damage. Yellow is lightly damaged, orange is moderately damaged, orange-red is heavily damaged, and red remains the color for a system that’s destroyed outright.

Version 1.06 (December 2014)

=====

This update introduced Lua scripting (see 5.3, Lua), differing turnaround times for aircraft (see 3.3.7, Air Operations), and unit level proficiency settings.

-Lua: This update introduced the first Lua scripting functionality into COMMAND. In this update, Lua was introduced as an “action” in the event editor. Scripts included spawning units, assigning/swapping missions, changing EMCON settings, and changing the weather.

-Surge and quick-turnaround times in air operations: Doctrine can now be set for either “Surge” (the previous default, rapid pace like at the start or climax of a campaign) or “sustained” (significantly slower, “one package a day” model) mode that will affect how long it takes to ready aircraft. Quick-turnaround has been introduced for certain loadouts that have both simple weapons (ie, usually unguided bombs/rockets, nothing that requires advanced setup/programming) and an inherently simple mission (ie, “fly out, bomb, repeat as necessary”). However, after a certain amount of flight time, a long cooldown period is needed.

The default quick-turnaround mode is for “fighters and ASW” aircraft, which allows for appropriate aircraft with simple anti air/submarine loadouts (that would realistically need to launch quickly) to deploy rapidly without unbalancing the rest of the scenario.

-Unit level proficiency settings. While side-level proficiency settings were added in 1.02, this enables it to be changed for specific units to simulate differences among individual crews/formations. Whereas in past versions of COMMAND, one would have to make two different allied sides (IE, “Iraqi Republican Guard”, “Iraqi Regular Army”) to have varying proficiency, this enables a scenario creator to have one individual side (IE, merely “Iraq”) and individual units varying from aces to novices.

-New feature: Satellite pass predictions. For all their invaluable strengths, one large, constant weakness of satellites has been their predictable paths. Selecting this function from the “game” menu and clicking at any point on the map will bring up a prediction window of how long it will take for the satellites in the scenario to reach that point and how long they will stay in sensor coverage once there.

-Patrol/Support altitude overrides. Aircraft on patrol and support missions can be ordered to fly at a specific altitude, either in transit or on station.

-Radars able to detect ground targets are now limited by target speeds. A stopped unit is harder to pick up on radar than a moving one.

-“Submarine Datum” contact: If an inbound torpedo is detected, a circular 10nm wide submarine contact is generated under the premise that the launching unit is close by somewhere. If a sub-launched missile is observed leaving the water, likewise a smaller 1nm sub contact is generated. This makes ASW more effective.

-Improved refreshing of the DB viewer window when different units/weapons are selected, preventing the need to constantly close and reopen it.

-Fixed a long-standing bug on the “unit enters area trigger” where the event would fire continuously whenever the unit was in the area. Now it fires properly when the unit “steps in”.

-Special Reserved Variable For Event Engine: UnitX. This is a global (scenario-wide) variable representing the active unit responsible for firing any of these triggers:

- Unit is damaged
- Unit is destroyed
- Unit is detected
- Unit enters an area
- Unit remains in area

So whichever unit causes of the above triggers to fire, it gets tagged as “UnitX”.

You can then use this special variable as a reference in any Lua-mapped function that takes a unit’s name/ObjectID as a parameter (so for example you can dictate “whichever unit enters this area will get assigned to this mission”).

- New Message Type: Mine Contact. Like all other message types, it can be configured to show a popup and appear on the message log. This specifically has to deal with naval mine contacts-a warning that one might have approached a minefield may be important.
 - Altitude parameters on a loadout's mission profile are now enforced for patrol and support missions.
 - Select New Homebase now works for all unit types, not just aircraft and groups of aircraft.
 - New event trigger type: Scenario is loaded. If the current scenario time is at or before the "start of scenario" date and time and the scenario is loaded, the trigger is activated. This is an alternative to the older "timed for one second after scenario start" trigger used for a similar effect.
 - Improved cloud and thermal layer indicators on throttle/altitude window. This displays a series of indicators at various altitudes indicated where the clouds/layers are. This helps for determining say, the right altitude to fly a laser-guided bomb strike at, so that the designator isn't blocked by the cloud.
 - Patrol change: Minimum # of units on station. This can be used to try and put a certain number of units on station at any one time. When it and the 1/3rd rule are both checked, the larger number takes precedence.
 - Additional factor for naval gunnery-ship size. A large ship will be more stable even in rough seas, while a small vessel will get tossed around more, limiting its accuracy.
 - Modified patrol behavior for aircraft. Aircraft will stop pursuing an identified target if they do not have the means to engage it. This solved the "Without being microed away, your MPA flies in circles above a hostile ship until it gets shot down" issue.
 - Torpedoes can be fired to max kinematic range. A new doctrine option (configurable for units and sides) allows torpedoes to be launched at their kinematic range. This can be set for manual shots only or AI engagement as well.
- (WARNING: NOT RECOMMENDED AGAINST ANYTHING EXCEPT A COMPLETELY UNAWARE TARGET. DON'T SAY WE DIDN'T WARN YOU)
- New ScenEdit feature: Clone unit. Unit customizations are copied along with the original class and name of the unit.
 - Added option to identify the selected unit/contact's image as a thumbnail on the right-side information panel.
 - Aircraft on RTB orders for reasons other than fuel exhaustion now accept course orders.
 - Pressing CTRL-X now copies the lat/long coordinates in decimal form at the current mouse cursor location to the clipboard. For example, pressing CTRL-X while the mouse cursor is hovering over Reykjavik, Iceland and then pasting will get the coordinates latitude='64.1305923461914', longitude='-21.8313770294189'.

This can be very useful for say, getting the right spot for a Lua script to spawn a unit later.

Version 1.07 (March 2015)

=====

This update introduced...

- Weapon release authorization. Players can configure the number of weapons launched against a specific target, the number of launchers in a salvo, and can limit the range at which the AI will fire a weapon (ie, so that they don't waste weapons in a launch that will almost if not certainly miss)

-A revamped mission editor. Compared to the pre-1.07 mission editor, the newer one enables flight size changes, “scrub measures” (to enable one-and-done strikes or to prevent increasingly smaller flights from pointlessly taking off to something they can’t complete), and the choice of straight-line or angled flight paths. See

<http://www.warfaresims.com/?p=3653>

-“Auto-bundled” scenario attachments. See <http://www.warfaresims.com/?p=3710>

-Updated UI functionality.

Updated functionality

- Ignore plotted course when attacking: moved to doctrine
- Hold Fire turned into doctrine setting and merged with Engage Unknown to become Weapon Control Setting (WCS)
- Added menu item to window for configuring Quick Turnaround for airborne aircraft
- Added menu item to window for setting fuel and airborne time for airborne aircraft

Doctrine changes

- Doctrine tab in Doctrine window re-done
- EMCON tab in Doctrine window updated
- Added Weapon Release Authorization (WRA) tab to Doctrine window
- Ctrl + Shift + F9 opens side-wide Doctrine window

Air Ops (F6) window

- Added 'Assign to mission' button to allow the player to assign aircraft to missions
- Added 'Doctrine' button to update various setting for aircraft whilst still on the ground
- Added context (right mouse-click) menu

Boat Ops (F7) window

- Window redone, added 'Abort launch' button and synched looks and functionality with AirOps
- Added 'Assign to mission' button to allow the player to assign boats to missions
- Added 'Doctrine' button to update various setting for boats whilst still in dock/davit
- Added context (right mouse-click) menu

Weapon Allocation ('Attack Target') window

The Weapon Allocation window has been re-designed. One of the most noticeable changes is the ability to multi-select targets and assign salvos to all with one mouse click. This functionality uses the Weapon Release Authorization (WRA) table to automatically create fire missions with pre-defined weapon quantities. It should be noted that assigning salvos this way does not take into account shooter limitations or automatic fire range limitations. Pressing the 'Assign Salvo' button multiple times will add additional salvo quantities of weapons.

Another significant change is that the number of weapons fired or already impacted as part of the ongoing salvo are displayed alongside yet-to-fire (assigned) weapons. This means the player gets a much better overview of what targets are currently under fire and the weapons involved.

The updated window also allows the player to add and remove shooting units and targets. In theory, this means that the player only needs to open the window once per scenario, and add/remove units and targets as needed. Destroyed units and targets are automatically cleared from the list.

Finally, the player can now resize the window, and it will also auto-update when the player runs the simulator. The player can also de-allocate large quantities of weapons with one mouse click.

Mission Editor

* The "Area Editor" component (used on mission editor etc.) has a new button: "Highlight + center on selected". If one or more reference points are currently selected, clicking this button highlights them and brings the map camera to their geographic center. This is useful, for example, for quickly seeing and moving around the reference points that comprise a patrol area.

Contact management

* New UI feature: You can now filter-out a selected contact ("Filter-out" command on the contact context/right-click menu). Filtered-out contacts are shown ghosted and no information on their movement vector, sensor & weapon range symbols and AoU is presented on the map. This can be useful if you have a lot of contacts of no interest (e.g neutrals) on the map and want to eliminate the clutter they create. You can cancel a contact's filter-out status the same way.

You can temporarily override a contact's filtered-out status by hovering the mouse cursor over it. This allows filtering-out multiple contacts to reduce map clutter, and then checking them individually "at a glance" to determine if they're worth resetting their filter-out.

-Land Range for weapons: Weapons now have separate ranges for hitting land targets (ie, a naval gun that's limited by its fire control against moving sea targets but only by ballistics against stationary land ones) and additional range rings for the land range.

-AAW engagements are now heavily dependent on the target's speed and angle. Previously, an aircraft would get a full evasion bonus no matter what the context. Now, a missile aimed at a slower (by fighter aircraft standards) from the rear will have a realistically higher chance of hitting than one aimed at the side of a rapidly moving opponent.

-AAW missiles (ie, mostly naval SAMs) attacking large enough targets now disable proximity fuses and rely on their (considerable) kinetic energy to do the damage instead.

-Submarine battery recharge rate has been significantly increased, and the recharge rate is faster the slower the submarine is moving (the diesels can recharge the batteries and not drive the sub)

-AI Doctrine Improvements:

-Aircraft will use the tempo settings of their current bases unless overridden by unit/mission level doctrine settings.

-New "Weapons Control Status" replaces "fire at unidentified contact" and "hold fire". The three new settings are "Weapons Free" (fire at any non-"friendly" contact), "weapons tight" (fire at confirmed hostile targets) and "weapons hold" (only fire in immediate self-defense).

-Airborne aircraft assigned via menu to existing strike missions will have flight plans automatically generated.

Version 1.08 (July 2015)

=====

This update focused on UI and simulation improvements.

-New Weapon Types: Anti-torpedo torpedoes and related warning systems.

-New weapon type: Enhanced fragmentation warheads, intended as a modern alternative to cluster rounds.

-New weapon type: contact explosive (sabotage and suicide).

- New “ship” (aka “that’s no island, it’s an aircraft carrier”) category: Mobile Offshore Base. The hypothetical Mobile Offshore Base has been added to the database and is designed to accommodate any aircraft type that can fit on its runway, even one that isn’t normally carrier-capable (so a MOB can hold F-16s and A-10s, for example)
- Aircraft body IR signatures are now dependent on speed. This mainly affects aircraft that can move very fast (and thus have a lot of heat friction) for a sustained time.
- Bringing up the weapons window and clicking the “show arcs” checkbox now reveals the engagement arc of a unit mount.
- The manual weapon allocation window can now optionally present “soft” restrictions (ROE, etc..) which prevent auto-fire even if the “hard” restrictions (ability to see/illuminate the target) are in effect.
- Targeting vectors turn yellow when the unit is physically able to engage the target but not cleared to do so, and green when they are cleared.
- Added “reset all side scores” on Losses/Expenditures window.
- Coastlines and borders now fade out as the player zooms in on the terrain.
- Subs now always snorkel/recharge if at periscope depth or higher, and now handle their speed more intelligently (via approach speed and knowing not to chase targets they’ll never catch)
- Snap up/down (how fast and far they can move vertically) limits are now applied to AAW guided weapons.
- Side-firing (AC-130, etc...) gunships now maneuver properly to line up their weapons.
- UNREP is now performed at a more realistic pace, and fixes have been introduced to prevent overloading mounts on the ship being replenished.
- New side-level option: Can auto-track civilians. The selected side automatically sees and tracks civilian/commercial units. This simulates access to publicly available tracking sources, and can be useful for law enforcement scenarios or to keep units from pointlessly chasing obvious civilian contacts.
- Mechanical-scan radars can no longer conduct volume scans while guiding a weapon, going “blind” while doing so (this is why fighters operate in pairs!). (An advantage of phased array and ICWI radars is that they can)

Version 1.09 (October 2015)

This update accompanied the Northern Inferno DLC. New features included campaigns (5.4.2, Campaigns), EMP blasts from nuclear detonations that can destroy sensors, and new Lua scripts (5.3, Lua.)

- Campaigns. This allowed for the creation of campaigns like Northern Inferno. Players can now create a set of linked scenarios with a score threshold that must be met to progress to the next scenario *and* an ending description.
- EMP blasts. Nuclear detonations now produce an EMP blast. The effect of the blast does *not* depend on the yield of the nuclear warhead, but rather on its altitude and distance to the target. The state of the target system also matters. A turned-off system can still be affected but is less vulnerable. Extremely important is the generation of the system-early vacuum tube electronics are the most resistant, which each succeeding generation becoming less and less so.
- Lua: Two Lua scripts were added:

Add Explosion. This enables a warhead to be detonated in a certain place and altitude. This can be used to simulate everything from high-altitude EMP-inducing nuclear blasts to conventional explosions. The ID number of a warhead used for the detonation can be found in the “weapons” section of the database under “warheads”.

Change Unit Side. Self-explanatory, this allows a unit to switch between sides. It can be used to simulate something like a boarding party taking over a vessel.

New Map Features:

- Place Names. This toggleable option included countries, cities, and major geographic features.

- Contact Emission changes. Either all emissions (so an SA-2 site would display both the Spoon Rest search radar AND Fan Song FCR) or only fire-control related emissions (just the Fan Song). This feature was added to prevent clutter when viewing contact with large numbers of sensors emitting.

Version 1.10 (February 2016)

=====

This update included Steam Workshop support. It also included cruise missile waypoints (so they can go around enemy SAM sites or behind mountains to conceal themselves), improvements to the ECM model, weight-dependent aircraft kinematics, and improved “Cooperative engagement capacity” (where something other than the launcher can guide weapons).

- Special Actions: These Lua script actions can be created by the scenario editor and activated in normal play as a sort of gamble. For instance, a player can attempt a high-risk, high-reward commando operation or conduct a routine request for reinforcements.

- Steam Workshop: Now COMMAND scenarios can be uploaded to the Steam Workshop. To upload, simply go to the editor menu (in the Steam version, of course), and select “Publish Scenario To Steam Workshop.”

- Waypoints for cruise missile attacks. To use them, go to ‘weapons allocated by selected attacker(s)’ in the “engage targets (manual)” window, press the ‘plot course’ button, and then ‘draw’ an appropriate path, pressing the F3 button when a satisfactory one has been created. The courses can be reset by clicking the ‘clear course’ button in the window.

- Improved ABM modeling. This update introduced changes to anti-ballistic engagements so that complex situations like the Patriot-vs-Scud engagements in the Gulf War could be readily simulated.

Ordinary SAM warheads now have only a 15% chance of outright destroying a ballistic missile target even if the endgame calc to hit it is successful. They have a 30% chance of causing a significant (CEP tripled) deviation of the warhead’s path. Otherwise, it will only cause a minor deviation (1.5x CEP).

Advanced SAM warheads include “Directional-frag” warheads that double the previous values (30% destruction, 60% significant deviation), and “hit to kill” warheads that, if they connect, will reliably destroy the missile every time.

Trajectory deviation multipliers for ABM engagements that don’t outright destroy the missile are cumulative. This means, particularly if the ballistic missile wasn’t the most accurate to begin with, that a series of hits that fail to completely destroy the warhead can still knock it away from its intended target.

- Aircraft kinematics affected by weight. Aircraft now have a “weight fraction” rating of the weapons and fuel load carried on them compared to the nominal, best-case scenario (clean and not much much). Even before other factors like crew skill, missile position, and altitude come into play, an aircraft with a higher weight fraction will be less maneuverable and thus less likely to avoid an attack.

-Cooperative Engagement Capability. What this means in short is that a unit other than the launching one can guide a weapon provided it has a proper datalink. To elaborate, it rapidly expands the possibilities, turning weapons into more than the sum of their parts. An E-2D with CEC datalinks can aid greatly in fleet defense, as it can look down and cover a far larger area than warships blocked by the horizon, terrain, or both.

-Improvements to electronic warfare modeling. Older-generation jammers are less effective against modern radars and vice versa. Phased-array radars (especially AESAs) are particularly effective at avoiding being jammed.

Version 1.11 (May 2016)

=====

This update introduced pier operations for long term scenarios (see 3.3.8, boat operations), more advanced options for aerial doctrine (ie, to enable being cautious on fuel or “fire one salvo and run” attacks), the ability to spot enemy aircraft/ships in their host bases, advanced submarine doctrine, and further refining of aerial refueling doctrine.

-Expanded “Winchester” and “Shotgun” weapons states. Aircraft can do everything from launch one long-range salvo and return to base to stay around and fight to the last cannon round.

-Expanded “Joker” and “Chicken” fuel states that trigger fuel-based RTBs other than the previous “Bingo” fuel state.

-Pier Operations. Ships can be rearmed and repaired at suitable bases now, although it takes considerable amounts of time for all but the smallest ships. Because of this, a “withdraw/redeploy” settings window has been added to determine when they go out and when they go back (there are more than a few historical occasions of not-fully repaired ships being pushed back out because of time constraints).

-Hosted units (aircraft/ships) at bases can now be observed by other sides if they’re in an open facility like tarmac spaces or are spotted taxiing to a closed hangar.

-Massive improvements to air-to-air refueling mechanics. See <http://www.warfaresims.com/?p=4190>

-Major update to submarine mechanics. Air Independent Propulsion systems have gotten a major rework, now being only able to keep up with battery series at creep throttle. Many new submarine doctrine features have been added, such as:

-Dive when threat is detected (with the intensity of the “threat”) configurable.

-Recharge battery %, transit/station-choosing the point when the submarine snorkels to recharge its batteries.

-Recharge battery %, offensive/defensive-as above, only when the submarine is actually engaged in combat.

-AIP usage. Allows the player to limit AIP usage, including saving it for combat.

-Dipping sonar. Enables unassigned helicopters to dip their sonar if they’re at the right altitude. Disabling it means they’ll only dip if manually ordered or on an appropriate mission.

-Avoid contact-selectable options are “yes” and “no”.

-Significant changes to replenishment rates, both UNREP and in piers.

-Aircraft may now jettison their ATG stores under attack when a doctrine setting allows for the sake of maneuverability under fire.

-Refined pathfinding engine. There’s an optional “fine-grain” navigation configuration that’s more precise but highly CPU-intensive.

-Unguided bombs and rockets dropped in salvos now use a more realistic “stick” method of landing. In previous versions of COMMAND, a large volley of bombs would “splatter” across a wider area, each one being considered its own calculation. After this update (and in V2), they will fall in a line near the area of the first impact. The dispersion of the

weapons still can vary based on factors like altitude, the proficiency of the dropping aircraft, and the technology level of the bombsight on the dropping aircraft.

This means that an aircraft unleashing all its unguided weapons in one volley can only truly attack one aimpoint at a time, meaning area damage is dependent on more bombs (there's a reason for the monster potential bombloads on classic aircraft like the A-6) and that a stick of bombs dropped from say, an old bomber with a rusty crew from high altitude against an airfield can easily (and realistically) miss it altogether and do no damage to the target.

- Changed how facility aimpoints, especially armored ones, are affected by blast damage.

- Changed cluster bomblet effects-due to having many small explosions, they are now much more likely to cause fires and inflict component damage.

- Units can no longer perform component repairs if they have more than minor flooding or fire. The crew is too concerned with trying to keep the ship from sinking or exploding to worry about the radar dish with disconnected wires.

- Only minor component damage can be completely fixed while a ship is underway. More serious repairs can only be conducted in a port.

- Due to the introduction of pier operations, "Unlimited Weapons At Airbases" realism setting is now changed to "Unlimited Base Weapons", covering ports (ie, reloads for missile boats) as well as air bases.

- The "Docking ops" window now has additional columns to display fuel, damage, and weapons. It also has color-coding to determine a unit's readiness compared to doctrine (an AI unit will only deploy if it's all "green", but it can still be overridden manually by a human side).

- Terrain-following "above ground level" (as opposed to "above sea level") settings are now available for aircraft under the throttle/altitude window. This enables units with terrain following radar capabilities (ie, F-111s) to do their thing.

- When a mission fails to launch aircraft, a logged message explaining why is now generated.

- New "Game Speed" options window to disable features that cause slowdowns.

- Mine deployment depths are now displayed in the DB viewer.

- Area validation-all windows with area editors now have a "validate area" button to see if the polygons are valid or not. All areas in a scenario are automatically validated on load, and a warning is given if invalid polygons have been made (this happens often with older scenarios made in early versions of COMMAND).

Version 1.11 SR1

=====

- Significant improvements to stick bombing mechanisms. The type of aircraft and number of bombs now has an effect on spacing between individual bombs.

- All weapon impacts (hits and misses) are now briefly shown on the map.

- When using the "zoom to the cursor position" option, after zooming, the mouse cursor is re-centered at the map.

- Keyboard tweaks. Spacebar instead of F12 now pauses the game. | shifts to the next unit.

- COMMAND can now load and run much larger scenarios with over 10,000 active units.

-Fixed an issue where an autosave failing due to memory exhaustion crashed COMMAND. Instead, a popup is displayed notifying the player of the failure.

-Improved support for high-DPI desktop resolutions.

-Recording tweak: Maintain God's Eye enabled/disabled preference between frames during playback.

-You can now enable/disable non-navigation zones and exclusion zones, in two ways:

a) Through the UI, as the grids on the relevant windows have an extra "Is Active" checkbox for each zone entry.

b) Through Lua. Usage example:

```
mySide = VP_GetSide({Name="US"})
```

```
myZone = mySide:getnonavzone("AAW Zone 1") [Alternative for excl. zone: myZone =
```

```
mySide:getexclusionzone("Forbidden Zone 1")]
```

```
myZone.isactive = "True"/"False"
```

Inactive no-navigation zones are not rendered on the map at all. Inactive exclusion zones are rendered with a dash line (instead of solid).

Version 1.11 SR6

=====

-Wholly revised right column UI, intended to be compatible with as many DPI settings as possible.

-"Damage control" button and flame damage indicator now enabled for aircraft.

-Detected jamming emissions are now in yellow text and do not fade.

-Active sonars now have a "bottom bounce" mode.

-Mobile facility contacts now have the same expiration time (2 in-game hours) as surface contacts.

-Ship AI tweak: Do not adjust speed for cavitation when on a simple plotted course.

-Weapons that impact on a set of coordinates now check if a target is actually there.

-Aircraft and aerial weapons that travel at high-supersonic/hypersonic speeds now get a massive (if linear) boost in their IR signature.

Version 1.11 SR7

=====

-New map/UI feature-directional unit icons. Can be selected by choosing the "directional stylized" option under "map display".

-Revised "realism options" window. The features are now more visible and cannot be changed in normal play.

-The unit magazines window now lists weapons by non-empty/empty first, and then alphabetically, so that the player can quickly see what weapons are available.

-The "New Contact" message type is supplemented by more focused types devoted to individual types of unit (land, air, surfaced, and submerged). This allows certain contact appearances to be more "noticeable" in message log settings (ie, a submarine that's the entire point of the scenario in question can raise a clock-stopping popup when detected, while a flurry of land contacts caused by an aerial recon run over an enemy division doesn't overload the log)

-Ships now maneuver to unmask the preferred mount for firing (try it with a Perry frigate with its mid-mounted gun).

- Non-aircraft units can now use escort logic (currently only for strike missions)
- When creating an “airborne” salvo (e.g. ballistic missile firing RV), if the target is an aimpoint, assume we’re firing at an actual target contact, and that is the one nearest to the aimpoint.
- Reimplemented incendiary weapons. The effect of an incendiary weapon depends on both the target’s size and armor levels.
- Gun rounds now lose kinetic energy the farther they travel, with corresponding effects on their damage and penetration.
- Detection checks now take sensor min/max altitudes into effect. (Many air-search radars have low detection altitudes, for instance).
- The localization logic previously used for ASW aircraft prosecutions is now used for any non-aircraft unit searching for a known but imprecise contact.
- Added loiter ability for certain tactical decoys.
- Large number of new and updated Lua methods.
- Lua commands and responses (including errors) now saved to a log file.
- Improved Lua error handling. Errors are displayed and execution stops when running in console mode. When running from special-actions, the function returns NIL but the script continues to run. The error is shown in the Lua variable ‘_errmsg_’ and the function in ‘_errfnc_’

Version 1.12 (May 2017)

=====

This update accompanied the Chains Of War DLC. New features include comms disruption (see 10.7, Comms Disruption), aircraft damage (see 5.4.9, Scenario Features and Settings), lasers, railguns, and EMPs.

- Major new feature: Comms Disruption. This enables realistic cutoffs from the network. (awaiting new revision)
- Major new feature: Aircraft damage. Now aircraft (when the feature is enabled) are no longer destroyed by any successful hit. Larger and/or more heavily armored aircraft are more resistant to damage. Damaged aircraft that return safely to base can still require long periods of time to repair-frequently longer than the remaining time in the scenario.
- Major new feature: Cargo. Now amphibious and air-landing operations no longer need to be “simulated” with either teleportation events or Lua spawns. Cargo operations can be performed manually or as part of a cargo mission.
- New weapon types.
 - Lasers
 - Tactical EMP
 - Railguns/HVPs
- New 2x time acceleration setting, in case real time was too slow and 5x was too fast.
- Improved support for 4K/UHD screens.
- “NO-CAV” is now displayed on the unit status panel for a unit deliberately slowing down to avoid cavitation.

-Addition to passive sonar model-Sonar masking. A quieter unit can use the noise generated by a louder one to hide. This is, unsurprisingly, less effective against more advanced sonars.

- Lua: Fix manual/desired speed on SetUnit()
- Lua: Enhance EndScenario trigger/action
- Lua: Enhanced "Unit Detected" event trigger

Version 1.13 (October 2017)

=====

This update added chaff dropping, changes to AAW/ABM mechanics (with more advanced missiles being more capable and taking intercept angles into account), and a new Lua console which serves as the base one in VNEXT.

-New Feature: Tactical Chaff Laying. Under "attack options", a unit can be ordered to lay down either a single bundle of chaff (single burst) or keep going until it runs out or is ordered to stop (continuous stream).

While disruptive to radar, chaff has its limitations which led to its relative decrease. First, aircraft must be *in* rather than simply behind the corridor, and, when faced with 3D radars, must take care with their altitude so that they're at the same level as the cloud.

Chaff is also most effective against older and/or lower frequency radar sets. A modern AESA will simply ignore it completely.

-New feature accessible from the throttle/altitude window: Sprint and drift for single units. The set speed is now the *average* speed, with the AI speeding up and slowing down as necessary.

- Isolated-POV (on-grid status) tweak: Visual indicators (dashed rectangle) for which of the displayed contacts are actually detected by the selected unit.

-Changes to AAW mechanics with detailed aircraft damage. Planes with a pressurized cabin are instantly destroyed if they have fuselage penetration, there's a chance the flight controls will be destroyed (dooming the aircraft) in fuselage hits, anti-missile engagements now have reduced chances to hit in oblique/crossing engagements, and AAMs with advanced fuses can do more than their nominal amount of damage.

-New V2 Lua console. This serves as the Lua console in COMMAND 2.0

Version 1.14 (February 2018)

=====

This update accompanied the Silent Service DLC. It includes contact detections on the right sidebar (so the player can see what detected a certain contact), a "do not cavitate" option for ships or subs wanting to stay quiet, and reattack capabilities for torpedoes and certain missiles that initially missed.

-New UI feature: Recent contact detections. Contact recent detections, ie. a list of the most recent successful sensor detections on a given contacts, showing who detected it and how. Example: <https://i.imgur.com/aEmfJnq.png> Clicking on the sensor name brings up the DB viewer page of the carrier platform and highlights the sensor stats. Clicking on the detector platform name selects the platform and centers the map on it. This can be quite useful in situations where it is not clear which of multiple units actually detected a contact (e.g. a surface group tracking a target) or in cases where it is desirable to understand which of multiple different sensors can actually hold the contact (quite common e.g. in submarine & ASW operations).

-New feature: Mark Contact Position, accessible by right clicking on the contact and selecting "mark contact position". It's useful for determining possible locations of uncertain contacts tracked over a long period of time.

-New Throttle/Altitude Window feature for ships and submarines-avoid cavitation. If enabled, the AI crew will move slowly enough to prevent noisy cavitation. It is ignored if the ship is avoiding a weapon (after all, it's already found and engine noise is the least of its worries)

-Users can now create circular areas of reference points in addition to the older rectangular ones. With CTRL-right click, there is now a "define area - circle" option in addition to the "define area - rectangle"

-New feature: weapon re-attacks. All torpedoes with terminal guidance and several modern ASMs have reattack capability (if they have it, it is marked as such in their database entry). Reattacking is possible if they are successfully jammed/decoyed just prior to impact, or if they fail their hit-or-miss check. In the latter case, the weapon must also pass a reliability check, with the chance of a dud being $\frac{1}{4}$ of the original PoH.

(For example, a torpedo attacks with a PoH of 80% and misses. The weapon can re-attack, but must first pass a reliability check with a 20% chance of failing ($80\% * 0.25$). If the check fails, the torpedo is a dud. If the check succeeds, the torpedo can proceed to re-attack either the same target or anyone else it acquires.)

If the weapon is eligible for re-attack, it passes right over/under the target and begins a spiral search until it is able to re-acquire a target (or runs out of fuel).

-New feature: Torpedo wire breaking. If a submarine conducts a "crash" turn (1.25 times faster than its standard turn, used only for weapon evasion) or is moving faster than ten knots while firing the torpedoes, there's a chance of the guidance wires breaking and the launched torpedoes becoming autonomous.

-New feature: Cargo pickup. Currently works only with aircraft (helos and VTOLs), and is manual-ordered only (no mission-driven behaviors as yet). To perform it, right-click on a suitable aircraft and select "Select unit(s) for pickup". The cursor will change to a targeting crosshair and you can select one or more target units from which to pick up cargo-type mounts, if they have (for example, to order a helicopter to pick troops from a mobile ground unit). Once it has picked up cargo, the aircraft can be ordered to RTB, and upon landing it will unload the cargo to its host unit.

-New feature: "Flaming datum". When a ship is hit by a completely undetected torpedo, an underwater "probable sub" contact is generated at the point of impact, giving any available ASW units a place to start looking.

-Ferry missions can now be used for ships and submarines.

-Improvements in submarine damage model. Now only shaped-charge warheads (on certain modern torpedoes) can penetrate underwater submarines. Normal explosives do more realistic shock damage from detonating close to the hull.

-New ScenEdit feature: "Chance of appearance" (Right click on unit in ScenEdit --> Scenario Editor --> Chance of appearance). This allows setting easily the chance of a unit actually showing up on the next scenario load (at normal-play mode). Units that fail the appearance check are discarded permanently. Applies to platforms only (not weapons).

-When exporting facilities using .inst files, boats berthed in these facilities are retained. This makes it easy to preserve nominal fleet ORBATs for specific historical periods (e.g. which ships were berthed in Norfolk on June 15, 2006).

-Min/Max target range for ships/subs/facilities/land-units on strike mission (e.g. "intercept and engage only targets within 200nm of your position"), similar to the previous "tethers" used for aircraft at their bases.

-The mission editor can now define how many aircraft in a patrol will investigate & engage contacts, as well as wingmen behavior. This is to prevent "contact pileups" where, for example, a single unidentified aircraft would enter the patrol/prosecution area, causing every fighter on patrol to just drop everything and scream towards it.

-Configuration files (*.ini) are now automatically consolidated at the [CMANO]\Config folder for easier management.

Version 1.15 (March 2019)

=====

This update included tweaks to visual/IR sensors to make them more realistically imprecise, a dynamic order of battle window, a personal persistent map option (use at your own risk), and changes in BVR doctrine (see 3.3.13, Doctrine)

-Personal Persistent Map Settings:

As we previously saw, the various map information and visibility settings (which layers to display, which range rings/wedges to show, misc. information such as targeting and illumination lines etc.) are set individually for each side in a scenario. This makes sense from a scenario design viewpoint, since each side may have different priorities depending on its overall operational tasking (e.g. prefer underwater sensor and weapon rings if conducting ASW operations). This does mean, however, that if a player prefers to apply his own custom settings at the start of a scenario, he has to do this at the start of each new game session. (*Settings are automatically persisted & applied when resuming from a save to continue an ongoing session*).

This situation has led to one of the biggest user complaints/requests, ever since the v1.0 release of Command, to be *"I am tired of setting my own custom map settings every time I start a scenario anew. Can we get an option to persist my custom settings and automatically apply them into any scenario I start playing?"*

We explained why this could be a bad idea (for example, it's an excellent way to miss potentially important information that the scenario author has laid out for the player's attention), but the request remained constant. So we obliged, and it is now possible to do precisely that.

Let us see how this works:

1. To begin with, set up your map preferences just how you like them - range rings, targeting & illumination vectors, datalinks, reference points, sonobuoys, datablocks etc.
2. Open the "Game Options" window, go to the "Map Display" tab. There, you will notice a checkbox labelled "Use personal map preferences" and a button labelled "Save current map profile as personal":
3. Clicking on "Save current map profile as personal" will make the current active map profile your personal, persistent one.
4. Checking the "Use personal map preferences" will enforce usage of your persistent map profile. From now on, whichever scenario you load and whichever side you switch to, the map profile settings in effect will be those of the persistent one.

To stop using the personal persistent profile, simply uncheck the "Use personal map preferences" checkbox. The map display will immediately revert to using the settings that the scenario author offers as default for the scenario & side at hand.

-Dynamic Order Of Battle Window. The "List by group" page of the ORBAT window is now dynamic, ie. it automatically refreshes as units are added or destroyed, take off, land, deploy or re-dock to their parent, join or leave a group etc. This makes it possible to keep the ORBAT window continuously open as a "running tally" (and quick-jump source) of friendly forces during a scenario, instead of a static snapshot.

-IR and visual sensors can now only generate precise contacts at short range, removing the old, unrealistic feature where a visual spotter could see an aircraft and instantly know its exact position.

-Finally resolved the long-standing issue of "blind AMRAAM shots":

- Units firing AAW ARH weapons (e.g. AMRAAM) must now actively detect the target on their own radar before being able to fire the weapon (ie. it's no longer enough to receive offboard contact from e.g. AWACS and fire based purely on that). This makes it much more challenging to employ AMRAAM-class weapons against stealthy targets or in a heavy-OECM environment.

- After launching an AMRAAM-class weapon, the unit must still keep detecting the contact in order to provide mid-course guidance to the missile. If radar contact is lost for more than 5 seconds, the missile goes into a "blind" state and flies straight ahead (similar to SARH weapons who lose guidance). (Appropriate AI/EMCON logic has been added to encourage this).

-Note that CEC-capable weapons such as AIM-120Ds *can* be launched "blindly" provided there is another appropriate unit to guide them.

-BVR Engagement Logic doctrine settings.

This changes the doctrine aircraft will follow when engaging with long-range missiles. The three choices are as follows:

-Follow missile straight-in: The aircraft will continue to follow the missile straight towards the foe. This can be used to simulate a unit with rigid doctrine, poor training, and/or a desire to close to visual range quickly.

-Crank if possible (DEFAULT): The aircraft will "crank" by tilting to the edge of its guidance radar and slowing down. This symbolizes a desire and training to stay at long range if possible.

-Crank and drag: The aircraft will not only crank but will, once the missile goes active and its guidance is no longer necessary, "drag" and move away from the target. This symbolizes a greater desire to stay at long range at any cost as well as the skill to do so.

-Updated South China Sea terrain data.

-The diesel "noise penalty" for snorkeling submarines is now gradual and dependent on the speed of the submarine. Not surprisingly, stationary subs are very quiet while rapidly moving subs are very loud.

Version 1.15.6

Major changes to MAD (Magnetic Anomaly Detector) sensor modelling.

-- MADs are no longer precision sensors, they have bearing ambiguity and absolutely no range estimate (the range estimate is essentially "anywhere from right in front of us up to max sensor range")

-- The practical MAD detection range is affected by the displacement ratio (target displacement vs max reference figure) and under-keel depth (shallow is worse, because of the bottom background)

-- MAD scan rates have been increased (only in DB3000 v478+). This rewards faster sensor platforms (e.g. P-8 vs P-3 vs helicopter) as they are able to cover more distance per given time and thus refine the contact position through triangulation faster.

Version 2.0

=====

COMMAND Version 2.0, or Command: Modern Operations, is the official sequel to Command, containing a completely revamped engine.

User Interface

- General user interface: The entire UI has been renovated on a dark theme with emphasis on improved usability
- New, high-resolution map layers:
 - Sentinel-2 Cloudless (2016): <https://s2maps.eu/>
 - BMNGv2 (improved version of NASA's Blue Marble NG)
 - Relief-90: High resolution land-only overlay
 - Stamen Terrain (<http://maps.stamen.com/terrain>)
 - Stamen Roads & Cities (<http://maps.stamen.com/terrain-lines>)
 - Terrain land-cover type (Example: <https://i.imgur.com/NNmb2Wx.jpg>)
- Message Log 2.0: A radically improved message reporting system.
 - Additional unit-level message log: All messages relevant to a specific unit are aggregated on a window specific to this unit and only optionally also shown on the side-level message log. This both declutters the main message log and makes it much easier to quickly examine a unit's message history.
 - Message balloons: When an event happens on the main map, a "balloon" containing the message pops up for ~10 sec. This makes it immediately obvious where the event described is happening. The balloons appear at a random bearing and semi-random distance from the reported event, to avoid overlapping on top of each other. They stay visible for 5 seconds, then fade out.
The appearance or not of the balloons is configurable per message type (Game Options → Message Log).
 - The side-level message log groups reported messages by type, and highlights the unread ones. This makes it easy to prioritize reading important messages (e.g. weapon engagements, battle damage etc) and quickly catch up on unread traffic: <https://i.imgur.com/mpVkJF6.png>
 - When clicking any of the messages on the interactive message log, the map will auto-center on the message origin location.
 - Messages are type-color coded, both on the message log and in the balloons. Example: <https://cdn.discordapp.com/attachments/285833706273243138/451754006797352960/xFzgVmF.png>
 - When used in detached mode, the interactive message log can also switch to the "classic v1" raw-text style and back again, by clicking on the "Switch to [other style]" button: <https://i.imgur.com/G2Eof7L.png>
<https://i.imgur.com/H5PBcpu.png>
- Players no longer need to manually download and install the unit image packs; images are now automatically downloaded on-demand and displayed when a respective unit is selected.
- The map engine has been re-written almost from scratch, to offer a fast, smooth-flowing pan & zoom experience.
- UI addition: A series of shortcut buttons are available under the main menu, to provide quick navigation to most common commands and windows. These can be toggled on/off by the "Show/Hide Shortcuts" button: <https://i.imgur.com/tT8KhD0.png>

- UI addition: The “Evaluation & Scoring” window now has a “Scoring Graph” tab, where the player can witness the change of the side score over time, along with the reason for the changes. Example: <https://i.imgur.com/ulmYdux.png>
- UI addition: throttle/altitude shortcut buttons on right column: <https://i.imgur.com/kojiBTA.png>
- New UI feature: Contacts list on the ORBAT window: <https://i.imgur.com/iPNPEu6.png>
- UI addition: “Edit hosted boats” and “Edit hosted AC” shortcut buttons in unit status panel. These are visible only in ScenEdit mode (and as applicable, e.g. a platform will no air facilities will not display the “Edit hosted AC” button), and allow you to directly jump to the relevant windows: <https://i.imgur.com/Jd7o1VR.png>
- UI tweak: When selecting a group member on the group-status panel (for a mobile group only, not land installation), instead of displaying the DB entry for the unit, the map now selects and centers on this unit and switches to unit-view if necessary. This makes it easier to quickly jump into a specific unit from group-view.
- UI addition: Keyword search on ORBAT window. This works similarly to web browsers, ie. as you type the desired keyword, entries matching are highlighted: <https://i.imgur.com/NLI3ZaW.png>
- Map/UI addition: Show list of currently selected friendly units grouped by class/type, similar to Homeworld. Example: <https://i.imgur.com/F2eJqKj.png>
- UI addition: It is now possible to have multiple windows of the DB-viewer open concurrently
- UI tweak: The throttle/altitude window (aka F2 window) more clearly distinguishes between having selected a waypoint of a unit’s plotted course or the unit itself.
Example with waypoint selected: <https://i.imgur.com/2Sfo3nS.png>
Example with the unit itself selected: <https://i.imgur.com/zNMyuBX.png>
Also, by using the white “previous” and “next” arrows next to the unit/waypoint description, it is possible to select the previous/next waypoint or the unit itself without leaving this window.
- New UI feature: The right column (unit status etc.) can now slide in/out of the main window.

How it works: There is now a collapse/expand button next to the right column: <https://i.imgur.com/v2rR8xj.png>

Clicking the button will cause the right column to slide out of view: <https://i.imgur.com/z4nhUpv.png>

Clicking the button again restores the right column as normal.

This can be useful if the left side of the main window is taken up by UI elements (e.g. message log & 3D-view) and this way the player can use more map area.

- New UI feature: Time-step buttons: <https://i.imgur.com/Lx7MdCQ.png> . Clicking on any of these (while paused) will advance the simulation time forward by 15 sec, 1 min, 5 mins or 15 mins, and then stop again. The key combinations Alt + 1/2/3/4 are mapped to the 15 sec, 1 min, 5 min and 15 min timesteps respectively. This makes it possible to control time-flow without having the mouse cursor leave the area of interest.
- Map/UI tweak: When clicking on a stack of units/contacts on the map, a new menu pops up allowing selection of the specific desired unit. Example: <https://i.imgur.com/2ZTADQn.png> . This is designed to avoid the need for extreme close-up on a map location just to select a unit.

- New UI window: Line of sight (LOS) Tool. The LOS tool offers a new capability for visualizing sensor LOS of a given platform or an arbitrary geopoint.
To use it, select the desired platform, then on the main menu, click on Editor LOS Tool. The LOS-Tool window will appear, and the shaded area representing the platform's current LOS will be rendered on the map.

There are various settings we can change on the tool:

- Max distance (nm): This is the maximum distance for which the LOS coverage will be calculated. Increasing this can have a major impact on map-refresh performance.
 - Target Alt (m AGL): This is the assumed target altitude for which the LOS coverage is calculated. This can be very useful if we want to visualize LOS against, for example, an airborne threat at a specific above-ground altitude ("show radar coverage against an F-111 terrain-following at 60m"). If the "Same as observer" checkbox is ticked, then the LOS calculations assume a target at the exact same altitude as the selected platform.
 - Horizon type: This is Radar/ESM by default but may be switched to Visual instead. The EM horizon is somewhat longer than the strict visual horizon and this affects LOS coverage as well.
 - Shade Color: The color of the shade coverage can be altered, to avoid interference with other map elements.
- New map feature: If you zoom low enough to a naval vessel (ship/sub), you can see its proportional size at it would be really seen at that camera altitude. Example: <https://i.imgur.com/JjEmgGR.png>
 - New map feature: Aircraft range rings. Like with weapons, rings that show the theoretical maximum range of a given aircraft can be enabled or disabled.
 - Map tweak: When displaying an area/course-based mission (on map) also display the mission name.
 - Map tweak: Long lines (e.g. intercontinental plotted courses) now follow the correct great-circle (ie. curved) path.
 - New UI feature: Minimaps. There are 3 types: Global (entire planet), Scenario (fits to the scenario's units) and Camera-Following. Different types can be used concurrently. For example: <https://i.imgur.com/XrorOAJ.png>
 - New UI option: Full-screen view for main window (View --> Full-screen view), ON by default. The shortcut key for toggling this is **Alt+Enter**. This setting is persisted across game sessions.
 - New UI feature: Hover-info box. If the user holds down the CTRL key while hovering the cursor over a unit, group or contact, a box pops up with summary information on the object of interest. Example: <https://i.imgur.com/qbDdlc9.png>
 - New UI feature: Open recent scen/save files: <https://i.imgur.com/IETZiu.png>
 - UI tweak: Steam Workshop scenarios are now accessed on a separate tab in the "Load Scenario" window: <https://i.imgur.com/0y9sonw.png>
 - Added support for real-time visualization using Tacview (**IMPORTANT NOTE:** To use this feature, you must purchase Tacview Advanced or Enterprise edition) . How to use:
 - 1) Purchase & download Tacview (<https://www.tacview.net/>) and install.
 - 2) The "Game Options" window a dedicated "Tacview" tab, for setting the path to the Tacview executable. Make sure you set the correct path in order to make it work. (The path is saved together with the other game preferences)

- 3) Load up a scenario as usual, and then click on View --> 3D View. A secondary window should appear, with Tacview's default top-down view. You can focus on any unit, either by clicking it on the Command map as usual, or by Ctrl+LMB on the Tacview window itself. The mouse scroll zooms in/out just like Command, whereas using mouse-grab you can rotate the camera around the focused unit.

Tacview is fairly sparse in models and terrain out of the box, but can be significantly enhanced with add-ons.

- To add extra models: First, construct a suitable 3D mesh and place it in the proper folder (see here for instructions: <https://www.tacview.net/documentation/3dobjects/en/>). Afterwards, edit the Excel file [CMANO]\Resources\Tacview\Associations_DB3000.xls and add a new row in the relevant sheet with at least the DBID and mesh file name of the object. Command will then read the entry and provide the filename to Tacview to render the object. (This currently possible only for DB3000 units)
- To add more detailed elevation data & textures, see instructions here: <https://www.tacview.net/documentation/terrain/en/> . We are currently putting together a global-scale SRTM3 dataset (elevations) but imagery layers are a tougher proposition.

The window hosting the Tacview display has the same common properties as other CMANO secondary windows, so it can be moved around, resized etc. Two example configurations are:

Single monitor: <https://i.imgur.com/PqnPYJm.png>

Two monitors: <https://i.imgur.com/O7sjbS7.jpg>

- New UI feature: Personal persistent map profile (ie. maintain the same map preferences throughout different scenarios & sides, effectively overriding the scenario author's default settings).

How it works:

- Set up your map preferences just how you like them - range rings, targeting & illumination vectors, datalinks, reference points, sonobuoys, datablocks etc.
- Open the "Game Options" window, go to the "Map Display" tab. There you will find two new UI elements: <https://i.imgur.com/MyKYwpp.png>
- Clicking on "Save current map profile as personal" will make the current active map profile your personal, persistent one.
- Checking the "Use personal map preferences" will enforce usage of your persistent map profile. From now on, whichever scenario you load and whichever side you switch to, the map profile settings in effect will be those of the persistent one.

To stop using the personal persistent profile, simply uncheck the "Use personal map preferences" checkbox. The map display will immediately revert to using the settings that the scenario author offers as default for the scenario & side at hand.

- UI tweak: The "contact status" panel now also display's the contact's WRA-type information: <https://i.imgur.com/0vPkYJh.png>
This helps to better understand AI decisions on WRA-driven weapon allocations (e.g. "why are my planes shooting 8 instead of 2 Harpoons at this ship contact?")
- New map option: Colored datablocks. Configurable through "Game Options --> Map Display". When enabled, a unit/contact's datablock will have the color corresponding to its posture (e.g. red if hostile).

General gameplay & content

- New gameplay mode: Quick Battle Generator (aka "Command MadLibs"). The QBG offers players a way to quickly jump into a battle without much preparation or wider geopolitical factors. The player selects the template of the quick battle he wants to create, alters the settings provided by the HTML page, and clicks on "START".

Quick-battle templates are stored in subfolders inside the [CMANO]\QuickBattle folder. Each template consists of a simple HTML page and a matching Lua file.

"Behind the scenes", the values of the named HTML elements in the template pages are translated to matching Lua variables with the same name, and provided to the template's Lua script file which is then executed, generating the quick battle.

The framework is constructed specifically so that players can make their own templates and share them with others, just like with regular scenarios.

- The features previously available only to "Chains Of War" and related DLCs (Comms disruption, detailed aircraft damage, cargo/ambip/airdrop operations and advanced weapon types) are now integrated into the base game and available to all players.
- New music soundtrack and sound effects.
- Brand-new scenarios by Paul Bridge (Sirius) and other community contributors.
- All original official v1.x scenarios have been remastered by Rory Noonan (Apache85) to take advantage of new v1.x and v2.0 features.
- An extensive range of tutorials (some new, some carried over from Command v1.x) to ease in new players into the intricacies of modern operations. There are now three full tutorial sets for each major branch of operations (air, surface, underwater), plus dedicated tutorials for specific aspects of the game.
- The default map icon style is now "Directional", to present new players with more familiar icons representing various unit types. The "NTDS/APP-6" and "Stylized" type options remain available.
- Includes the latest versions of the DB3000 and CWDB databases.

Simulation & mechanics

- Simulation performance & scalability has been significantly enhanced. This is most evident in large or complex scenarios.
- Major new simulation feature: Terrain types.
The different types of terrain are now accurately represented, using NASA's MODIS Land-Cover 2012 dataset
(More details: https://lpdaac.usgs.gov/dataset_discovery/modis/modis_products_table/mcd12q)
The various terrain types have different effects on the mobility of ground units, on weapon effects (especially blast and frag warheads) and spotting visibility. In summary:
 - Water is impassable
 - Wetlands reduce speed to 10% of original (bog!), and impair visibility slightly.
 - Snow/Ice reduce speed to 20% of original, and impair visibility slightly.

- All forest types reduce speed to 30% of original, significantly hamper the destructive effect of blast and fragmentation detonations, and sharply reduce spotting ranges.
- Shrublands reduce speed to 50% of original, slightly reduce the blast & frag effects, and moderately reduce visibility.
- Croplands and vegetation mosaics reduce speed to 70% of original, and slightly reduce visibility.
- Open and wooded savannas reduce speed to 80% of original
- Barren / sparsely vegetated terrain and grasslands have no impact on speed, weapons or visibility.
- Urban & built-up terrain increases speed to 110% of original, but severely block both weapons and visibility.

The land-type information is displayed on the map-cursor blackbox, next to terrain slope info.

- New optional scenario realism option: Realistic Sub Comms. When enabled, submarines who go below 40m depth go off the communications grid. The player can, at any time, send a "bell-ringer" ELF signal to a no-comms submarine to recall it to re-establish comms (right-click on sub icon, click on "Summon to re-establish comms"). The sub will attempt to rise to shallow depth to rejoin (this may not be immediate, as it will try to evade nearby hostiles). Once it rejoins, the sub will share its contact updates with its parent side.
- Land units can now resupply from appropriate units (ie, ammo trucks) similar to UNREP-capable ships.

The unit can be ordered to resupply via the "replenish/refuel if necessary" command on the right click window. Once the supply vehicle is found/directed to, it will pull up alongside it and stop, with weapons being gradually transferred over.

- Land units now perform pathfinding during their navigation. The movement cost takes into account both terrain slope (pre-existing) and land-cover type (new). The combination of these factors generates the navigation route.

This is an example, with an M60 tank platoon at southern Attica ordered to move to the northeast of the peninsula: <https://i.imgur.com/VtqRzVZ.png> . The unit selects the displayed path both to avoid the surrounding mountain ridges (red ellipses) and to maximize its mobility, going through the urban area instead of the croplands to the east.

- Significant change in unit AI logic: The "evaluate targets" and "evaluate threats" logics are now not performed on every pulse, but instead on regular intervals dictated by the OODA-Targeting (modified by crew proficiency) and OODA-Evasion values respectively.

This has two effects:

1. These two OODA values, and crew proficiency, become even more critical to a unit's effectiveness and survivability.
2. Because these AI parts are called less frequently (and they are among the top CPU-killers), overall sim-core performance in large/complex scenarios is significantly improved.

- IR and visual sensors can now generate precise detections only at short ranges.

Specifically:

- Visual sensors make a precise detection only at distances shorter than twice the max classification zoom level of the sensor (e.g. if an aircraft can be visually classified at 5nm by a

visual sensor, all detections at longer than 10nm will be imprecise). The range ambiguity at long range is 10% of the detection range (e.g. if a detection is made at 20nm, the range ambiguity will be +/- 2nm)

- Non-imaging IR sensors make a precise detection only at distances shorter than the max classification zoom level of the sensor (e.g. if an aircraft can be classified at 5nm by anIRST, all detections at longer than 5nm will be imprecise). The range ambiguity at long range is 50% of the detection range (plainIRSTs are really bad at providing range estimates).
- Imaging IR sensors are significantly more capable than their non-imaging counterparts. Like visuals, they generate a precise detection at distances shorter than twice the max classification zoom level, and their range ambiguity is 20% of the detection range.
- Aircraft and surface-launched missiles now use the same improved pitch kinematics as air-launched missiles (ie. smoother climbs/dives).
- Lofted AAW missiles now begin their terminal dive earlier, to avoid a too-steep approach to the target.
- NEW: Improved, cost-based pathfinding logic. Ships now plot more realistic courses around islands, straits etc. and avoid "surfing" (coming too close to shore unnecessarily).
- Numerous tweaks to aircraft flight model, specifically for "combat" conditions. For example, aircraft no longer "wiggle" between headings as they must first roll towards the turn direction before committing to a turn. This in turn makes roll-rate much more critical to close air combat maneuvers. This is easier to observe in the Tacview window.
- New radar model factor: Frequency agility. Frequency-agile radars get two major benefits; they are more resistant to noise jamming and they are significantly less affected by doppler notching (see below). Mechanical-scan radars that are frequency agile have this flag on their DB values. All PESA and AESA radars are assumed to be frequency-agile.
- Doppler notching is now actively used as a maneuver, both for missile avoidance and for general radar detection avoidance. Aircraft can attempt to fly perpendicular to an emitter using doppler filtering in order to hide inside its "blind" velocity gate. The effectiveness of the maneuver varies with crew skill (an "ace" pilot will execute it far more effectively than a novice), to discourage manual micromanagement. Aircraft under missile attack with a doppler radar guiding the missile will also actively try to beam the radar instead of the missile (the geometry of the two axes can vary significantly). The maneuver is ineffective against pulse-only radars and less effective against frequency-agile radars.
- Air combat AI improvement: Aircraft now consider approaching fighters/interceptors as imminent threat, not just missiles. This helps AI-controlled aircraft perform more proactive evasive maneuvers against fighters about to perform gun attacks on them (e.g. MiG-17 vs F-105).
- New air combat feature: Pilot G-tolerance (a.k.a. "how to resolve the "fighters pulling hard turns indefinitely" realism flaw").
When an aircraft is considered to be performing "combat maneuvers", an extra UI element is shown on unit status panel: <https://i.imgur.com/odY2E8O.png>

This represents the crew's tolerance to hard maneuvering. The longer the aircraft is continuously pulling a hard turn, the more this buffer fills up. (Getting out of a hard maneuver, e.g. reversing a turn, reduces this strain. This is one of the reasons that scissors are a popular practice in RL dogfights.)

Once the tolerance is exhausted, the crew begin suffering G-LOC and have to significantly relax the turn (this is easier to notice in Tacview, but you should also observe a noticeably larger turn radius on the top-down view) while regaining their senses and control of the aircraft. Of course the aircraft is particularly more vulnerable during this recovery period.

-New patrol mission option: Repeated loop. This allows units on patrol to take exact patterns (similar to support missions) when not engaged offensively.

- When a unit re-joins the comms grid, the sim now provides much more finegrained feedback on what type of information was updated for a contact (if any), and skips mentioning contact merges that produce no new info.

Scenario Editor & Scripting

- New ScenEdit feature: Merge scenarios into a single one (Editor --> Merge Scenarios). Scenario #1 is treated as the "base" for the merge, so behind the scenes you are essentially cherry-picking all transferable items from scen #2 and adding them to scen #1 (the log output also makes this clear). Elements that are "atomic" or otherwise hard to merge (date & time, weather, title & description etc.) will, on the output, be those of scenario #1.

Version 2.1 (Command Modern Operations V 1.01)

=====

- A significant number of new official Tacview models included (see 10.8, Tacview).
- Terrain type legend added (see 6.5, Map Settings Drop Down Menu)
- 15x time acceleration added (see 3.1, The Globe Display)
- Numerous speed/performance enhancement features, as well as bug fixes.
- Tacview no longer displays all units of all sides automatically, adding a proper fog of war mode.
- Major ABM performance enhancements.
- Changed the shade of red displayed to stand out more easily against dark backgrounds.
- Mission AI tweak: For non-specific mission types, the default "start looking for tankers" fuel threshold has been changed from 60% to 30%. This helps alleviate cases where aircraft nearing their strike targets abruptly turn back and start looking for a suitable tanker to refuel.
- In-game sounds/music now have adjustable volume (see 6.4.4, Sounds and Music)
- Numerous database updates.

Version 2.2 (Command Modern Operations V 1.02)

CMO v1.02 Release Notes

=====

* Additional significant improvements to mine warfare:

- Map shows the connection (a line) between the mine and who is clearing it.
- The sweep from a ship starts offset from the unit location, to simulate the sweep hanging back from the stern. This pushes back the mine explosions as they are caught in the sweep.
- MCM gear can have a delay factor to when it can become active.
- Improved actions for MCM-oriented torpedoes like Seafox, so they can look for new mines if they miss the initial target mine
- Pathfinding on mine-clearing mission is improved so as to not to spend too long finding paths. If a mine is detected in the path to the next waypoint, the path is shortened rather than recalculated. This gives a better flow to the mission.
- Units can be given a set of waypoints to follow, and the unit will attempt to follow this path while diverting to clear any mines within range.
- Mine-hunting torpedoes will be automatically fired against bottom and rising mines (which cannot be swept). This compromise has been made because a unit might fire the whole magazine against the first lot of mines detected, despite being able to sweep them normally. This does not stop the user from manually firing the torpedo using standard BOL.
- MCM gear is not immediately available when a ship turns unless the turn angle is small; this simulates the disruption to the sweep as it straightens out again (one cause of previous issues with sweeping was that the sweep was always active even during sharp turns, which meant a ship could set off multiple mines as it swang around, causing lots of damage to

itself).

- Aircraft OTOH quickly correct the sweep, but they fly an arc which can put them out of position for the mine they are after. They will try to self-correct and aim closer to the target mine; however this may not happen immediately. As the sweep is active, they could sweep mines that fall under their sweeps so the multiple attempts to line up the mine is counter by this.
- Ships will tend to back away from any other ships performing mine clearing operations. The sweep will be 'turned off' in these cases so as not to set off mines in that location.

Even though the mine clearing mission has been cleaned up to help minimize damages, this is a risky business for ships and severe damage can still happen.

- * Improvement to 4K-Fix tool (prevents null-reference exception).
- * Added support for CWDB in the Quick-Battle Generator. You can now specify which DB to employ in a QBG template, by using a 'hidden'-type HTML element. Examples: `<input type="hidden" name="DB" value="DB3000">` or `<input type="hidden" name="DB" value="CWDB">`. This element must be present in the Form.html file in the template folder. If it is not present, the QBG defaults to using the DB3000 database.
- * Added: [Lua] VP_GetScenario method (Exposes some scenario data such as duration, start time, complexity & difficulty, DB file used etc)

(Command Modern Operations V 1.03)

v1.03 - Build 1147.23 Release Notes

=====

Major changes to sonar model, full description here: <https://ftp.matrixgames.com/pub/Command ... HANGES.pdf>

Summary of changes:

- The thermocline layer is generally weaker then before for a given location, and completely stops being effective at 70+ degrees north or south latitude.
- The minimum bottom depth necessary to achieve converge zone (CZ) propagation is sharply increased per given location, and is no longer static but dependent on latitude. CZs are now also impossible at 70+ degrees north or south latitude.
- Most under-layer sonars (SOSUS/SURTASS excluded) lose the 2x detection range bonus afforded by the deep sound channel. This sharply reduces the in-DSC detection ranges of VDS & towed arrays.

* Certain IRST/FLIR systems can now produce AAW fire control-grade data, allowing EMCON-silent engagements with AMRAAM-class weapons (relevant discussion: <https://www.matrixgames.com/forums/tm.asp?m=4886491>). Note that the fire-control data generated by advanced IRSTs can be used not only for direct onboard shooting (e.g. F-18E uses IRST-21 to silently shoot AMRAAMs at target) but it can also be distributed to other platforms via CEC (e.g. F-18E silently tracks target, hands off to CEC-equipped ship which silently shoots SM-6 at target). These are the sensors in DB3000 v488 that have this ability: <https://i.imgur.com/ZclxXYD.png>

* ADDED: Visual and IR checks are now also susceptible to look-down clutter. For example it is easier for an IRST (or the plain Mk1 Eyeball) to pick out an aircraft over the horizon line than against the surface background.

* Datalink + TARH AAW missiles (e.g. AMRAAM, SM-6 etc.) can now go active without immediately going autonomous (this allows such missiles to be guided with greater precision & reliability against VLO targets or in strong OEMC environment). The datalink is cut instead only when the missile acquires the target (or impacts on it, still under positive control from the link provider).

* TWEAK: MineLaying changes:

1. Don't drop mines until the first WP is reached rather than drop as soon as hits area.
2. Allow a path to be set from the first WP so a pattern of mines can be laid. No path results in the current random pathing.

- * Numerous sim-core performance tweaks.

Other changes

- * ADDED: #14463: LOS tool with feet as an option
- * ADDED: #14495: "4 weapons" WRA option for difficult targets (see this thread for background: <https://www.matrixgames.com/forums/tm.asp?m=5000304>)
- * TWEAK: [Lua] "UpdateUnit" method can now handle addition/removal of magazine components (just like comms/sensors/etc)
- * ADDED: Setting "PauseOnDBView" (allows not pausing sim when opening DB viewer)
- * TWEAK: [Lua] Added game mode and state to "Scenario" wrapper
- * TWEAK: Color code differences in Doctrine/ROE:
 - The ROE label in RED means that the ROE differs to the next lower level. Example: a mission ROE option has a setting, but the child group ROE option differs.
 - The ROE label in YELLOW means that the ROE differs to the next upper level. Example: a mission ROE option has a setting, but the parent side ROE option differs.
- * ADDED: Lua functions to return UTC time / date from epoch
- * DB-Viewer tweak: Explicitly note non-radar AAW-FC capable sensors as such
- * FIXED: [Lua] SetMission.TankerFollowsReceivers flag not recognized
- * Performance tweak when selecting a single unit.
- * Performance - Significant performance increase when selecting multiple units.
- * TWEAK: Unit speed/alt window: Do not display layer information for subs if layer strength is 0
- * ADDED: Lua missing Navigation doctrines
- * ADDED: #14341: Added UI buttons to quickly enable/disable all range indicators
- * TWEAK: #14041: Database viewer tweak for fuel records
- * TWEAK: #8534: Cloning facility/ships now also clones their hosted A/C and/or boats + 2 new cloning functions
- * TWEAK: Renaming is now a bit more scalable & FIXED a conflict with renaming of RPs
- * TWEAK: Added exit Hotkeys to secondary windows
- * TWEAK: #13571: Streamlined Renaming of Units
- * TWEAK: #13802: On the "Load Scenario" window, remove ".scen" in the name of each scenario in the selection tree (Leave .save extension to differentiate)
- * TWEAK: The muzzle flash of a firing unit is now proportionate to caliber & better rocket-plume detection & IR detection
- * TWEAK: The start screen is now full-screen or maximized (ie. taskbar visible) according to the preference of the main map window
- * TWEAK: #13181: Tweaked chaff effectiveness against early ASHMs (more gradual)
- * TWEAK: #14241: Towed arrays cannot operate at over 15 knots
- * TWEAK: "Pure Illuminator" radars (e.g. SPG-62) now considered FCR for emission ID purposes

- * FIXED: #14377: Ballistic missile battery fires all tubes when you only want it to fire one
- * FIXED: #14401 SA-11 shot wrong direction
- * FIXED/ADDED: #14499 - Pathfinding issue: units would eventually get stuck in sloppy areas. Improved how land units speed is computed given slope, terrain type, and the unit's traction system.
- * FIXED: #14314: "Log Element Lacking Summary" for Gods Eye View Event
- * FIXED: #14329: No-Nav zone visible from the other side of the globe
- * FIXED: #14332: AOR Target Subtype is "5025" in the Event Editor
- * FIXED: #14365: If you clone a unit hosting aircraft using Shift+c you get reference error
- * FIXED: #4355: ScenEnded trigger never fires
- * FIXED: #14279: [Lua] Problems with the PatrolMission Table
- * FIXED: [Lua] VP_Scenario not always getting correct file name/path
- * FIXED: #13552 - Placenames layer being drawn over
- * FIXED: #14414 - BM could no longer be shot down by Lasers

- * FIXED: NoNav zone import/export issues
- * FIXED: Exclusion zone import/export issues
- * FIXED: [Lua] Null-reference exception on scenario wrapper if there is no current side
- * FIXED: #14437 - Mobile Facility Not Stopping to Engage Aircraft
- * FIXED: "Object Reference ..." Error starting Quick Battle when TacView open
- * FIXED: Startup crash due to missing registry values when determining desktop DPI
- * FIXED: Wrong color on label "Auto-Evade"
- * FIXED: #14432: Lua Restoring Default Doctrine Settings.
- * FIXED: #14381 - Truncated Lon/Lat
- * FIXED: #14420 - Sprint and drift issue
- * FIXED: ABM DLZ calculations causing significant sim slowdown
- * FIXED: #13587: ship is overloading munitions
- * FIXED: #13668: UNREP message
- * FIXED: #14379: Re-naming issue
- * FIXED: #14354: Sprint & Drift doesn't work.
- * FIXED: #14345: Contact Report and Contact wrapper typedescription field
- * FIXED: Low-altitude cruise missiles released from high altitude dived too sharply in order to reach their cruise altitude
- * FIXED: #13904 - Nuke reliability issue
- * FIXED: #14407 - Main gun not reloading
- * FIXED: #14491 - Rare cases of broken saving process
- * FIXED: #14285: Lost focus on ORBAT window
- * FIXED: #14287: Ctrl+s shortcut stopped working
- * FIXED: Game Options --> "Game Speed" tab - Checkboxes not working
- * FIXED: #13338: "Pickup units" was no longer working
- * FIXED: #13475: Ships not pathfinding correctly when RTB
- * FIXED: #13332: FIXED group and formation editor
- * FIXED: #13322: Pause/Play shortcut not always working (bypassing stolen focus)
- * FIXED: #14234: ARM Targeting
- * FIXED: #10643: Improved Hotkeys for windows (Bring focus / exit)
- * FIXED: #10842: Minor string modification
- * FIXED: Glitched & empty save folder showing up in scenario loading UI
- * FIXED: #14039: Autodetected ship is listed as SKUNK when all details are available in contact report
- * FIXED: #13652: 361 degree scale map?
- * FIXED: #14040: Ship orientation not able to be set in ScenEditor
- * FIXED: #13986: RTB issue while in Nonav / failed bounce off
- * FIXED: #14046: Aircraft are having issues when ordered to RTB
- * FIXED: #13918: Cargo Pickups can't be canceled
- * FIXED: #14167: Cargo Mission tweaks for unusual mission areas
- * FIXED: #14223: Torpedoes run out of steam too early
- * FIXED: #13328: Esc & Enter keys not working to close window
- * FIXED: Flaw in Satellite Prediction calculations
- * FIXED: #14274: ABM calculation causes the game to freeze / crash
- * FIXED: #14233: Missile engages target behind it
- * FIXED: #13765: Holding down Ctrl-key for "Hover view" doesn't work when cursor already over unit
- * FIXED: #14251: Off-Boresight missile not working
- * FIXED: #14232: Allow to fire sequential salvo (additional) against the same target.
- * FIXED: #13884: Reloading during engagement for AAA
- * FIXED: #13984: Infinite Acquisition/Drop loop of friendly out-of-comms sub contacts
- * FIXED: #13985: Retained targeting of very ambiguous target for Gun, Rocket, Iron Bomb
- * FIXED: Label too small for text on "Edit Scoring" dialog
- * FIXED: #14143: Temporary signatures and muzzle flash
- * FIXED: "Canary's cage" old scenario duplicate + Canary's cage Lua fix

- * FIXED: #9273: Aerostats cannot dock
 - * FIXED: #13182: Problem with too-perfect SSM homing
 - * FIXED: #12437: Straight-running torpedoes DLZ issue
 - * FIXED: #13672: Units not firing allocated weapons
 - * FIXED: #10936: LCM trying to find path inland
 - * FIXED: #10841: All ships can unload cargo onto shore
 - * FIXED: #14209: Too much information from emissions off non auto-detectable buildings
 - * FIXED: #14024: BOL-Launch bug
 - * FIXED: #13983: Engaging targets in No-Nav zones
 - * FIXED: #13646: All tubes reload, but only three were fired?
 - * FIXED: #14228: CVN/AOE UNREP Woes
 - * FIXED: #14047: Poor map performance when group selected (and datalinks visible)
 - * FIXED: #14098: LOS tool inconsistent
 - * FIXED: #13971: LOS Tool Results Still Show up After Dialog Closed
 - * FIXED: #13420: Scrolling Air facilities list using mouse wheel is slow
 - * FIXED: #13321: Detection emission information shadowed by datablock
 - * FIXED: "XXX" text for unknown contact speed/altitude displayed on the same line
 - * FIXED: #14217: Weird AI A/C behavior in Down Town, 1967
 - * FIXED: #14135: 'Opportunity target' Doctrine ignored if 'investigate outside patrol area' setting is off
 - * FIXED: #14173: A/C RTB after BVR shot without providing guidance
 - * FIXED: #14177: Message log position creeps upwards on main window
 - * FIXED: Adding Mount to Boat Hosted in a Facility Adds Mount to Facility
 - * FIXED: #12998: AC wont attack heavily damaged target from mission strike list
 - * FIXED: #14245: Strange missile guidance behavior
 - * FIXED: #14144: Out of Comm A/C friendly contacts are continuously dropped / Reacquired
 - * FIXED: #13966: Changing Database clears scenario without warning
 - * FIXED: #14475 - Mounted OECM should reference to the platform when displaying the detecting platform
 - * FIXED: #14481 Strike Missions Range Warnings when no target is defined
 - * FIXED: #14438 - Flight xyz RTBing immediately... message spam
 - * FIXED: #14459 - Spawned Aircraft could have a negative AGL altitude
 - * FIXED: Cargo mission not clearing Items when removing motherships
 - * FIXED: #14458 - LUA issues involving posturing via LUA Code involving Unknown Contacts
 - * FIXED: #14364 - Crashing JASSM & JASSM-ER
 - * FIXED: #13757 - CMO Nuclear Weapons Issues (one aimpoint mount in target facility unaffected by blast, torpedoes not destroyed by sea surface blast, BOL launch of ICBM targeting sea ignored weapon minimum range)
 - * FIXED: Autosave fails if the scenario is untitled
 - * FIXED: #14023 Added - Autosaves no longer get chain deleted after an exception
 - * FIXED: ScenEdit_UnitY() nil on Unit Destroyed
 - * FIXED: #13984 - Infinite Acquisition / drop of out of comm subs
 - * FIXED: #14335 - Strafing won't work.
 - * FIXED: #14319 - ESM logic compares NATO vs ITU frequency bands causing some sensors to be blind (applies to AN/FLR-9)
 - * FIXED: Ghosted group members not being rendered as semi-transparent
 - * FIXED: Missing doctrine checks
 - * FIXED: Weapon unneeded maneuvering trying to intercept at slow speeds right after launch
 - * FIXED: Weapon terminal point set too far for high air closure target
-
- * Includes support for new "Kashmir Fire" campaign.
 - * Includes all updated Quick-Battle templates.
 - * Includes the v488 version of the DB3000 database.

(Command Modern Operations V 1.04)

CMO v1.04 - Build 1147.33 Changelog (changes from v1.03 - B1147.23)

=====

- * ADDED: New page on DB-viewer dedicated to sensors.

- * New map option: Automatically filter-out contacts based on posture: <https://i.imgur.com/PFRq52e.png> . This can be very useful to help declutter a very busy map (e.g. in Northern Inferno #1 once SOSUS detections start happening) without manually having to filter-out each contact. To enable the 'auto-filter', select the type of contact from the dropdown list and left-click; a tick mark will be shown if it is enabled. You can only select neutral biological and civilian contacts at the moment. If the contact turns hostile or unfriendly, it will turn off the contact's filter out flag.

- * ADDED: When a weapon record is multi-packed (e.g. 4x ESSMs on single Mk41 cell), display this information on DB-viewer and "Weapons" window (Example: <https://i.imgur.com/7JDIAJ2.png>)

- * ADDED: Aircraft baseline (colored green) and loadout-specific RCS signatures shown in DB-viewer.

- * FIXED: "Red-X" problem after resuming from OS sleep/hibernate/screenlock

- * Various SAR/CSAR improvements:

- Added CSAR role to aircraft (in addition to SAR)
- Check range to pick up
- Do pickup of cargo OR SAR-eligible unit (any unit could be so picked up leading to lost units)
- Eligible units need to have Lua ScenEdit_SetUnit("CSAR")
- The "unit" Lua wrapper now has a WasPickedUp property to indicate if it has been picked up

- * ADDED: #14548 Add UI option to edit ship/submarine/facility fuel levels: <https://imgur.com/C4o9yI7>

- * ADDED: "Sub-type" filter on DB viewer

- * ADDED: #14505: Fire and Flood should be able to be edited through the GUI

- * ADDED: #14547 - Different escort response distances for fighters and SEAD aircraft

- * ADDED: #14564: Mission Editor option: Avoid Cavitation

- * ADDED: #14564: [Lua] Avoid Cavitation option ("AVOIDCAVITATION" parameter on ScenEdit_SetUnit, plus "avoidCavitation" property on "unit" wrapper object)

- * ADDED: Arc-coverage information for platform mounts & sensors on DB viewer

- * ADDED: [Lua] Added 'unassign' keyword to SetUnit() and wrapper (Completing un-assigns the unit as if hotkey 'u' was pressed)

- * ADDED: [Lua] Added ship ready-time, as property on unit wrapper-object and also as parameter ("TIMETOREADY_MINUTES") on ScenEdit_SetUnit

- * Added UI tweaks to "Set Fuel Quantity and airborne time" window.

- * Added missing chaff-rejection effectiveness values for Late-1980s & Early-1990s sensors

- * ADDED: "Mine countermeasures" items to "Damage Control" window

- * ADDED: SBR now also records fuel levels for ships and subs

- * ADDED: [Lua] 'crew' property added to unit wrapper

- * ADDED: [Lua] 'pitch' & 'groundSpeed' properties to unit wrapper

- * ADDED: [Lua] Added methods ResetLossExp, ResetScore

- * ADDED: [Lua] Added property contact.lastDetections

- * ADDED: [Lua] unit wrapper 'jammed'/'jammer' properties

- * ADDED: [Lua] unit.airborne_v as numeric seconds

- * ADDED: [Steam] Ability to change the visibility of workshop items

- * Tweak: #14606: [Lua] Unit wrapper damage percent reliability
- * Tweak: #14672: Show warning on the weapon form if usable only by Command-PE
- * Tweak: #14698: Doctrine Inheritance on Patrol missions - "Ignore Plotted Course" setting (If doctrine is inherited and already set to 'yes', don't change it. This allows users more flexibility when controlling units under that mission)
- * Tweak: Force "DB migration" progress to use folder scenario loaded from (This will usually be where the matching .ini file is located)
- * Tweak: Steam Workshop load of scenarios
- * Tweak: When adding a/c to facility, check base facilities for appropriate runways
- * UI Tweak: Add scroll bars to Steam Workshop list
- * UI Tweak: If aircraft cannot launch due to no runways, log an "air ops" message
- * UI Tweak: On "Edit hosted aircraft" window, warn about aircraft/runway incompatibility
- * UI Tweak: Show unit being picked up
- * UI Tweak: Weapons window - show if no suitable datalink present on platform for the selected weapon (this can be very helpful when adding non-standard weapons on a platform in ScenEdit)

- * FIXED: #14493: Cannot manually change torpedo load
- * FIXED: #14500: ASW Patrol Missions would not drop sonobuoys in "Repeatable Loop"
- * FIXED: #14545: Ctrl+S hotkey not working in Scenario Editor
- * FIXED: #14552: Scoring graph not working as intended
- * FIXED: #14573: [Lua] SetUnit script with Unitname and Group both set to same (Can't change a group on a group)
- * FIXED: #14574: [Lua] SetUnit used with Group is unable to accelerate Group past Cruise
- * FIXED: #14576: Aircraft Ignore Profile Attack Altitude
- * FIXED: #14578: Avoid cavitation option for groups
- * FIXED: #14581: Unable to unload RHIB
- * FIXED: #14587: Weird behavior with SM-3 and ABM combat
- * FIXED: #14604: Side Wrapper "unitsBy" case of exception.
- * FIXED: #14613: [Lua] Misspelling "Aircraft" when adding aircraft locks up sim
- * FIXED: #14625: Aircraft missing #2 - clear assigned host if destroyed
- * FIXED: #14625: Aircraft missing - confirm group delete
- * FIXED: #14630: Restarting a campaign scenario restarts as a stand-alone scenario
- * FIXED: #14637: Can't clear Message Log in Interactive view
- * FIXED: #14649: Strike missions should allow altitude override when ingress
- * FIXED: #14677: AutoDetect change and Platform Name
- * FIXED: #14679: Missions with activation time can't be made inactive if there is no end time
- * FIXED: #14681: UNREP - not breaking off if unable to fit weapons in receiver
- * FIXED: #14683: Minelaying Issues (allow group members to drop mines and to use release height - see: <https://www.matrixgames.com/forums/tm.asp?m=5056506>)
- * FIXED: #14684: Air-to-Ground weapons fired outside vertical-plane boresight limit
- * FIXED: #14686: Increase the Number of Scenarios listed on Steam Workshop Update Selection menu
- * FIXED: AC-130 gunship not manouvering properly to fire its side guns
- * FIXED: Autosaves not saved correctly
- * FIXED: Bad loadout causes exception
- * FIXED: C-130 could pick up cargo from base while airborne
- * FIXED: Changes on "Sensors" window not updating "Damage Control" window
- * FIXED: Correct OOB contact-BDA text
- * FIXED: Crash on NI campaign start
- * FIXED: DB viewer scrolling on filters
- * FIXED: Docking clear assigned host if destroyed
- * FIXED: Fixed inconsistent error handling in SE_AssignUnitToMission()
- * FIXED: Flight plan errors reporting unable to takeoff on airborne flights
- * FIXED: Guided projectile was early-detonating before impacting on surface target
- * FIXED: Inactive no-nav zone affecting target determination

- * FIXED: No datalink status when datalinks exist
 - * FIXED: Out-of-memory error during ABM engagement
 - * FIXED: Presence of sea-mines borks trigger weapon subtype
 - * FIXED: Prevent exception if event-triggered action is removed during the action's own execution
 - * FIXED: Prevent the OS from going to sleep or turning off the monitor (similar to media-player programs).
 - * FIXED: Previous secondary windows were kept opened when loading a scenario through "Load recent" menu
 - * FIXED: Right column inherited doctrine settings not updating correctly
 - * FIXED: Severe performance hit in sat-to-sat LOS checks
 - * FIXED: Show evaluation gauge even if no thresholds set
 - * FIXED: Spawning weapon records in magazine
 - * FIXED: [A2AR] Refuelling A/C was not adjusting its throttle to his parent's speed while attached. Creating significant fuel consumptions discrepancies.
 - * FIXED: #14496: Unloaded guns considered 'underwater'
 - * FIXED: #14500: ASW Patrol Missions would not drop sonobuoys in "Repeatable Loop"
 - * FIXED: #14515: Scoring level counter Problem
 - * FIXED: #14556: Can't launch mission, because I need 2147483647 aircraft
 - * FIXED: #14559: Scenario won't generate a valid Delta File
 - * FIXED: ASW aircraft not flying towards the AOU of their primary target submarine.
 - * FIXED: Contact filter drops 'unknown' civilian types
 - * FIXED: Spawning weapon records in magazine
 - * Various fixes on minimaps
 - * Significant UI performance improvement in "busy" scenarios (lots of reported messages)
- * Includes updated versions of "Kashmir Fire" scenarios #1, #3, #11 and #12.
- * Includes support for the new "Red Tide" campaign pack.
- * Includes the updated versions of the DB3000 & CWDB databases. Also includes updated Tacview 3D-shapes and association files.

Build 1147.44 Release Notes

=====

- * FIXED: Spiral 3 Never Hits Ship
- * TWEAK: Laser designators can now guide all SALH weapons (even if the weapon does not have the designator listed as a compatible director)
- * TWEAK: The Tacview version checker now provides more feedback
- * FIXED: Land Cover Legend Mismatch
- * FIXED: Timing mismatch can lead to a unit's "Determine Primary Target" method never being run (this can lead to any number of issues related to targeting AI)
- * FIXED: #14740: Certain Side Functions nonfunctional after Side added via LUA
- * ADDED: Ability to open online Lua Docs standalone or on specific instruction
- * Includes updated versions of all official scenarios.

Build 1147.42 Release Notes (changes from B1147.34)

=====

- * **NEW FEATURE:** Radar resolution cell. (See this page for background: [https://www.radartutorial.eu/01.basics/ ...ll.en.html](https://www.radartutorial.eu/01.basics/...ll.en.html)). If multiple units are located within the resolution cell, the radar will detect only one of them. This feature confers an extra advantage to higher-frequency radar systems, as they are better able to distinguish between closely-clustered targets.

* **NEW FEATURE:** Radar vertical scan angle limitations. Radars can only look up/down to a maximum of 30 degrees. This restriction severely curtails their ability to search for targets high above or well under them, and has indeed been used and exploited operationally (Example: [https://theaviationgeekclub.com/the-sto ... 14-tomcats](https://theaviationgeekclub.com/the-sto-14-tomcats)). There is an exception, however: Phased-array radars can scan upwards close to the vertical limit. Satellites are also exempt from this check since they are designed for significant look-down angle.

* **NEW FEATURE:** Phased-array radars (PESA/AESA) suffer from gain degradation (hence effective range reduction) when attempting to detect targets far off the antenna boresight, both on the horizontal and vertical plane.

This schematic shows the coverage difference between a mechanical-scan set (APG-68(V)9) and two AESA radar options for the F-16: [https://belisarius21.files.wordpress.co ... eba-3.png](https://belisarius21.files.wordpress.co...eba-3.png) . Note the severe drop on high off-boresight detection range on the two AESAs because of the gain reduction.

In addition to the changes in the radar code, the difference is also displayed on the

map: <https://i.imgur.com/lB5BXn6.png> (The "flower" coverage pattern on the ship is because of the individual SPY-1 faces, each of which is a PESA antenna).

Aircraft with phased-array radars will now crank more conservatively (reduced angle off-target) to compensate for this effect.

* ADDED: [Event Editor] Checkbox to show all unit types in target triggers

* Tweak: Lower distance to check for localisation change

* UI Tweak: "Speed & Altitude" window - Missing waypoint type that might be causing exception

* TWEAK: "Ignore plotted course" setting limited to Patrol mission

* TWEAK: Tac-Nukes under semi-active guidance (e.g. nuclear SAMs) detonate if they go blind after travelling half their fly distance (ie. armed)

* TWEAK: #14380 - [GUI] Mission list ordered by alphabetical order

* TWEAK: Placing mobile facility in a difficult terrain shows a warning

* ADDED: [Tacview] Added support for the MaxFrameRate switch (only on Tacview 1.8.7+)

* ADDED: Lua wrapper for Scenario object allows setting scenario Title

* ADDED: #14339: Hotkey for "Define Area" (Ctrl + K)

* ADDED: #13784: GUI QoL 1.1 (Add New Mission)

* ADDED: #13787: GUI QoL 1.2 (Mission Editor)

* ADDED: Three QoL requests from forum:

- Able to sort planes/boats in the air operations/boat operations by ready time / loadout

- Ability to hide/show unavailable aircraft on Air-Ops window.

- Hangar/dock now show capacity occupied/total capacity (occupied percentage)

* Added menu button to delete waypoint and shortcut info

* Numerous tweaks & fixes on pathfinder logics, to address reported PF issues.

* The practical scenario size limit (in terms of number of units) for consistent saving has been increased.

* The integrity check on registered DB files has been improved.

* Placed more tactical-AI calculations to run in parallel execution. This should result in better sim performance and higher CPU utilization in large scenarios.

* The DB integrity check mechanism has been improved.

* FIXED: Changing mission creates unnecessary waypoint

* FIXED: Delete shortcut not working --- Deleting a waypoint was deleting units in some cases

* FIXED: #14202: Curse you ASAT operators! DLZ issues persist

* FIXED: #14472: Can't adjust columns in pop-up boxes

* FIXED: Vertical boresight issue with unpowered EO-guided bombs (fixes strike tutorial issue with KAB bombs)

* FIXED: Contact age-time discrepancy

* FIXED: It was possible to create multiple missions with the same name

* FIXED: Weapon vertical-plane boresight check was preventing pure-SALH LGBs from dropping from high altitude

- * FIXED: #14207: Delta Files not capturing changes to non-operational units (e.g. ships in port)
- * FIXED: Passive sonar masking checks can causes significant slowdown in scenarios with numerous underwater targets (e.g. Nothern Inferno #1)
- * FIXED: Single aircraft not obeying the "Ignore plotted course when attacking" setting
- * FIXED: Reset "Ignore plotted course" setting when on patrol and contact is lost
- * [Lua]: Skip function register if function with that name is already registered. (This prevents cases where a function is repeatedly registered and causes a performance issue)
- * FIXED: Some of the tutorial scenarios were referencing the original CMANO or COW-exclusive features. This has now been corrected.
- * FIXED: Torpedo "Impossible to intercept target" within kinematic range
- * FIXED: Sub not being structurally repaired on port
- * FIXED: #10978: Ship at pier not repairing bridge and rudder
- * FIXED: Added missing ship identifiers
- * Fixed: #14602: [Lua] "SetDamage" - damage points parse to decimal (instead of integer)
- * FIXED: #14734 - Missiles lose guidance in save file
- * FIXED: #14579 - AC change Altitude and Speed
- * FIXED: Aircraft Unload Cargo When Moving from Hangar to Flight Deck
- * FIXED: Issue w/ Weapon Release Authorization Settings
- * FIXED: #14732 - SLBM RVs circling the point where the target was
- * FIXED: #14584 - SAM Site fire doesn't account for prosecution zones
- * FIXED: Cargo mission - Won't un/load the right quantity of stuff
- * FIXED: cargo quantity to unload was reset to 0 when the "Cargo Mission" window was reloaded
- * FIXED: Weapon that was accurate enough to direct-hit target was not correctly airbursting on top of it
- * FIXED: #14675: Bombing in wrong place but still hitting
- * FIXED: #13811: GUI QoL 7.3 Order of Battle + Order of Battle Contacts
- * FIXED: #13801: GUI QoL 4.2 Side Selection and Briefing window
- * FIXED: #13800: GUI QoL 4.1 Side Selection and Briefing window
- * FIXED: #13816: GUI QoL 11.4 Throttle and Altitude settings
- * FIXED: Out-Of-Comms Group vector icon
- * FIXED: ORBAT - Contact default list is collapsed, and no tab 3 in this branch
- * FIXED: Can't kill that platoon and a lag problem
- * FIXED: [Lua] "Contact" Wrapper .fromside null-reference exception
- * FIXED: ASBM/RV "Hyunmoo 2C" salvo causes loop "Error at 100971"
- * FIXED: Chaff ineffective?
- * FIXED: Space bar re-opening doctrine screen
- * FIXED: List of missions should be in alphabet order, for aircraft on airfields
- * FIXED: #13784: Bugfixing for default Expand/Collapse state of ORBAT
- * FIXED: Airops form show "Label1" in the quick turnaround info when empty
- * FIXED: QOL [Feature Request] cmanouser1's list of QOL requests (1.4 1.5 3.1 7.2 7.5)

- * Includes all the scenarios of the Kashmir Fire campaign
- * Includes various fixed/tweaked scenarios (Operation Brass Drum, Shifting Sands #13, The Silent Service #2, Red Tide #3 & #10)
- * Includes new Tacview 3D models by KJohnston (A-12, EA-18G, Boeing ATS, SR-72, YF-23, S-300/400 TEL) and updated association sheets
- * Includes updated versions of available Quick-Battle templates
- * Includes the v492 releases of the DB3000 and CWDB databases

Build 1147.48 Release Notes

=====

FIXED: Fallback CEP for SASS weapons 'SASS weapons (SAM As Surface to Surface)

- FIXED: Aircraft don't fit airbases in v494+ DBs
- Added support for UAS aircraft-size categories
- FIXED: F-16 #4789 appearing as biplane?
- Includes the proper & tagged version of the Showcase #1 scenario (Showcase Queen Elizabeth.scen); the "QE2.scen" file from the previous build should be discarded.
- Includes updated URL for remote-downloaded images
- Includes several updated/fixed/tweaked official scenarios by Kushan:
 - o Raid on Kismayo 2013
 - o C-LIVE Kuril Sunrise
 - o Northern Inferno #9 - Fox Two 22 Sept 1975
 - o Shifting Sands #13 - Seconds out Round Two!
 - o Strike Tutorial - 4 - Laser Dance
 - o Submarine Tutorial 1.1
 - o Submarine Tutorial 1.2
 - o Submarine Tutorial 1.3
 - o Submarine Tutorial 1.4
 - o Submarine Tutorial 2.1
 - o Submarine Tutorial 2.2
 - o Submarine Tutorial 2.3
 - o Submarine Tutorial 2.4
- Includes new installation-import files by Kushan:
 - o US BMEWS & ABM sites
 - o China - South China Sea islands & reefs (as of July 16, 2022)
- Added Tacview meshes

Build 1147.49 Release Notes

=====

- Fixed #14827: Taiwan Blitz scenario freeze during BM attack
- Fixed: Order of Battle shows units of the last opened scenario
- Fixed: [B1268.2] Iowa Phase II Interdiction Ship aircraft error
- Fixed: Exception while reading terrain file may lead to sim freeze
- Fixed: Conformal fuel tanks should not be jettisoned

(Command Modern Operations V 1.05)

Build 1307.1 Release Notes (changes from B1307)

=====

- * ADDED: ATO, Flightplan Editor and Flightplan generation button into the "Unit" Tab of Mission Editor
- * ADDED: [Lua] Added EMCON to SE_GetDoctrine
- * ADDED: Added support for area opacity; Using the area & repoint manager you can now define the opacity/transparency level of the selected area/zone
- * ADDED: More informative feedback when failing to create a minefield or when an individual mine cannot be placed
- * TWEAK: [Mission Editor] Added vertical scroll-bars, so the lower panel can be accessed by users using lower resolution
- * TWEAK: Added ship restrictions check to landing similar to takeoff
- * TWEAK: Catapulted UAVs can be launched/landed from any open space on ships
- * TWEAK: Modified all the instances of objective time with Time on Target
- * TWEAK: "Intermittent emissions" now has a default value for emission and sleep durations

- * TWEAK: Area & Ref point manager now sorts RPs by alphabetical order
- * TWEAK: [Lua] SE_AttackContact:
 1. Allow course setting in mode 1
 2. If no mount specified, use first one with appropriate weapon DBID.
- * FIXED: Flightplans & Flightplan Templates.
- * FIXED: Ferry missions where not using Flightplans properly. Now TOT is now fully functional (for ferry missions) and is defined as the landing time of the flight.
- * FIXED: Performance Issues with B1304.1
- * FIXED: Repeated Fuel State Warnings
- * FIXED: Ops planner time-based "mission satisfied" discrepancy in elapsed time in double-flame mode
- * FIXED: Range of some unguided weapons not being shown properly
- * FIXED: Submarine not properly showing all fuel types in use
- * FIXED: Performance Drop when unit with no image selected
- * FIXED: Benign discrepancy in UI refresh when creating a new zone.
- * FIXED: Exception if player deletes last mount from landed detachment
- * FIXED/TWEAK: Shift+RMB menu now explicitly informs of no suitable weapon available
- * FIXED: "Sinking ship" contacts were invisible
- * FIXED: Incorrect orientation on Tacview MQ-1 model

Build 1307.2 Release Notes (changes from B1307.1)

=====

- * ADDED: DB viewer shows 'Supports WayPoints' on applicable weapons
- * ADDED: Unit WRA-type to DB Viewer
- * ADDED: "Cargo can self-deploy" option be for Cargo Delivery missions
- * RE-ADDED: #13801: GUI QoL 4.2 Side Selection and Briefing window
- * RE-ADDED: #13800: GUI QoL 4.1 Side Selection and Briefing window
- * RE-ADDED: #13816: GUI QoL 11.4 Throttle and Altitude settings
- * FIXED: Ferry missions where not using Flightplans properly. Now TOT is now fully functional (for ferry missions) and is defined as the landing time of the flight.
- * FIXED: Flightplans & Flightplan Templates
- * FIXED: Empty leader flightplan when reloading flightplan editor
- * FIXED: Blank WRA reproduceable
- * FIXED: Multidomain TOT was not working properly with HGVs
- * FIXED: [B1307.1] ATO edit flight plan Lambda exception
- * FIXED: "Omniscient" Side Awareness was not working
- * FIXED: Amphibious ground unit were not deployed after being moved from the cargo bay to the dock if they were the subject of a Cargo Mission that was requiring them to self-transport
- * FIXED: Amphibious units were not able to RTB in docks
- * FIXED: Mission Description in ME was truncated
- * FIXED: Incorrect orientation on Tacview MQ-1 model
- * Includes the new DB-description files by CV60.
- * Includes a new series of tutorial scenarios for the multi-domain strike planner, by Kushan.
- * Includes a corrected & updated version of Submarine Tutorial #1.3.

Build 1307.3 Release Notes (changes from B1307.2)

=====

- * TWEAK: Ship-borne UAVs with no dedicated launcher

- * TWEAK: Resized "Scenario Briefing" window to 80% of screen dimensions
- * FIXED: BVR missiles not lofting at short-range shots may bleed energy excessively
- * FIXED: Error message in Strike Planner tutorial #2
- * FIXED: Very slow game speed (exception in palletized weapon logic)
- * FIXED: Multi-Axis Attack method for stand-off strike missions don't work correctly (when using standoff A2G weapons)

Build 1307.4 Release Notes (changes from B1307.3)

=====

- * Various tweaks & improvements on Intermittent Emissions:
 - Fixed an issue preventing non-active sensor types to perform detection and tracking
 - Allowed permanent "silent" mode when setting emission value to 0
 - "Wake upon detection" mode performs much better
 - Fixed a display issue with negative value being shown after interval deadline
 - Toggling to intermittent now use the proper interval initial value
- * ADDED: Scenario/Save search ability in "Load Scenario" window: <https://i.imgur.com/T73qCmC.gif>
- * Added: [Lua] Method GetKeyStore() to dump to a table all the items stored
- * ADDED: [Mission Editor] Show show base name of targeted facility
- * ADDED: [Lua] air/dock facilities on unit wrapper
- * TWEAK: AAW missiles with turbojet propulsion are now considered constant-powered (e.g. AAW loitering munitions, "turbo-SAMs") and thus do not use the boost-coast model
- * TWEAK: Use proportional navigation for autonomous-guided weapon intercepts
- * TWEAK: "Add New Satellite" menu - country list sorted
- * TWEAK: Auto-create holding area for UAV Class 1
- * TWEAK: Launch Class 1 UAVs from ground units
- * TWEAK: Allow UAV with launcher to fly from any open air facility
- * TWEAK: Allow access to DB from ORBAT (mission and type tabs)
- * POSSIBLE FIX: Significant UI slowdown when a friendly unit is selected <-- **NEED CONFIRMATION - UNABLE TO REPRODUCE ON TEST**
- * POSSIBLE FIX: Game stalls after loading saved scenario <-- **NEED CONFIRMATION - UNABLE TO REPRODUCE ON TEST**
- * FIXED: Vertical-boresight check was incorrectly calculating the altitude difference between observer & target
- * FIXED: Operation planner would break if no side is defined.
- * FIXED: Contacts being lost due to floating point math rounding
- * FIXED: Exception thrown after setting TOT and clicking "Create flightplans"
- * FIXED: Evaluation if the TakeOffLocation moved was never executed, that could lead to a wrong TOT when AC depart from a ship
- * FIXED: Auto-generated flights for patrol missions do not get the flight planned assigned.
- * FIXED: #14898: MDSP Flightplans Broken and Other Issues
- * FIXED: [Lua] GetContacts were missing base contacts in some cases.
- * FIXED: [Lua] ScenEdit_SetUnit was not registering last known position of unit, this could cause invisible unit's icon or drawn in invalid position until the next refresh.
- * FIXED: area & ref point manager interaction with CEZ was no longer working as intended
- * FIXED: Error when trying to spawn in Venezuelan Satellites VRSS-1/2
- * FIXED: Interception/strike missions broken after war planner update?
- * FIXED: Helos being forced into UAV hanger when no other hangers
- * FIXED: Cross-thread issue for Cargo Editor, ORBAT, and Mission Editor when an unit is added/removed in the scen while the windows are opened
- * FIXED: Aircrafts on a strike mission refuse to launch at anything. They just circle the target

- * FIXED: Strike planner bug, disappearing aircraft
- * FIXED: Mouse-move order was no longer working for shift+mouse multi-waypoint course plotting.
- * FIXED: Submarines resetting depth at each waypoint
- * FIXED: Shift+RMB was a scen edit only feature (?)
- * FIXED: Magazines disappear when exporting/importing unit
- * FIXED: Importing a unit with extra parking facilities reports overfilled for every extra aircraft

Build 1307.6 Release Notes (changes from B1307.5)

=====

- * NEW SIM FEATURE: Per-sensor mast height (optional)

Apart from the existing ability to specify an overall mast-height for a platform/unit, it is now possible to specify the mast-height value per sensor. This can be useful in modeling the different height placement (and thus horizon coverage, and the now-critical ability to clear the local skyline, see below) for each individual sensor in a platform.

In the DB3000 v499, two ship platforms have explicit per-sensor mast-height values:

- * #2867 - DDG 51 Arleigh Burke [Arleigh Burke Flight I] -- United States (Navy), 2018, DMP, SM-3, AEGIS BMD
- * #122 - DDG 51 Arleigh Burke [Arleigh Burke Flight I] -- United States (Navy), 1991

* NEW MAJOR SIM FEATURE: Terrain type-dependent skyline / land-cover height. The simulation now clearly distinguishes between terrain elevation height (ie. the actual ground) and the "skyline" height outlined by either natural features (forest trees, plantation etc.) or man-made structures (buildings, bridges etc.). This is critical for a number of different sim features, from land close-combat (e.g. land forces need to poke their sensors above local features to get sufficient LOS) to low-level flight (some aircraft can fly between the trees/buildings, most cannot).

Several new land-cover types have been added to the existing ones. The overall list of land-cover types and their associated skyline-height values is as follows:

Water => 0
 Evergreen Needleleaf forest => 5
 Evergreen Broadleaf forest => 5
 Deciduous Needleleaf forest => 5
 Deciduous_Broadleaf forest => 5
 Mixed forest => 5
 Closed shrublands => 1
 Open shrublands => 1
 Woody savannas => 5
 Savannas => 1
 Grasslands => 1
 Permanent wetlands => 1
 Croplands => 1
 Cropland Natural Vegetation Mosaic => 1
 Snow And Ice => 1
 Barren Or Sparsely Vegetated => 1
 Urban And Built Up => 9
 Urban - Close Inner City => 30
 Urban - Spaced High Rise => 45
 Urban - Attached Houses => 6
 Urban - Close Industrial => 9
 Urban - Spaced Apartments => 20
 Urban - Detached Houses => 6

Urban - Spaced Industrial => 9
Urban - Shanty Town => 4

Custom Environment Zone UI changes and new land-cover Types

The Custom Environment Zone window UI has been re-arranged, with sonar environment overrides grouped together on one side and land-cover overrides grouped on the other. You can now set a custom land cover height (skyline), which will override the default land cover height for the CEZ:

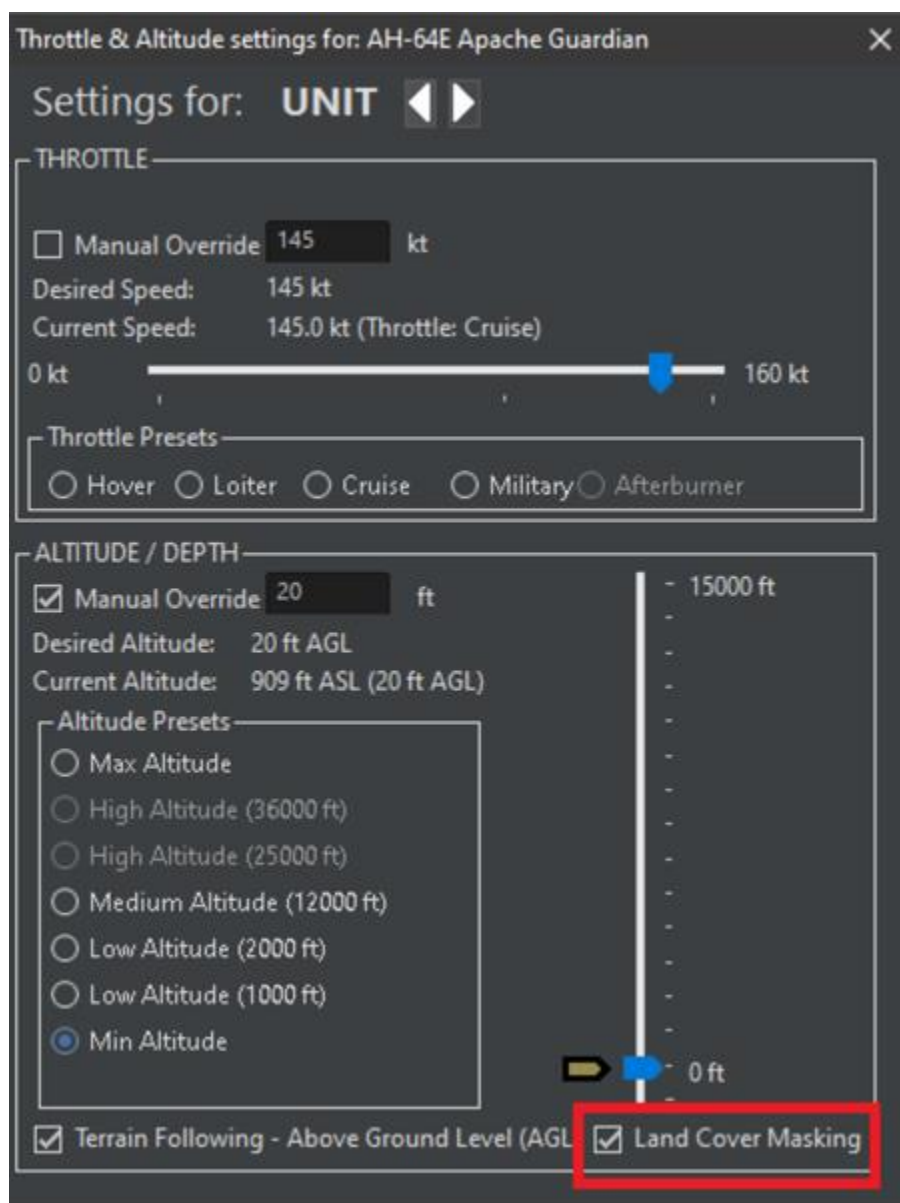
The screenshot shows the 'Custom Environment Area Editor' window. At the top, it says 'Weather modelling level: Level 0'. Below this, there's a 'Level 0' tab. The main area is divided into two columns. The left column contains 'Area Name' (Flat Land), 'Average temperature:' (15 °C), 'Rainfall rate:' (No rain), 'The sky is:' (Clear), and 'Wind / Sea state:' (Calm). The right column contains corresponding sliders and labels: 'Heavy storm', 'Full with thick clouds', and 'Hurricane'. Below these are two sections: 'Sonar Conditions' and 'Land Cover'. The 'Sonar Conditions' section has a red warning 'WARNING OVERLAND RP DETECTED' and a 'Thermal Layer' section with 'Ceiling', 'Floor', and 'Strength' sliders, all set to 0. There's also a 'CZ Interval' slider set to 0. The 'Land Cover' section has a 'Cover Type' dropdown set to 'Use_Underlying_Values' and a 'Set Custom Height' checkbox with a value of 0 meters.

The cover type 'Use_Underlying_Values' allows you to create a CEZ that doesn't override land cover at all (for example if you only wanted to override weather and/or sonar conditions.) If you do select a specific land cover type, you can then check the 'Set custom height' checkbox to override the cover height for that CEZ. The height value is in whole meters.

Aircraft 'Land Cover Masking' Flight Mode

'Land Cover Masking' (aka "fly between the trees/buildings") is a new flight mode for aircraft. Only aircraft with the DB capability flag for 'land cover masking' set will be capable of this flight mode (DB v499+ required). In DB3000 v499 all rotorcraft (incl. some UAVs) have this capability.

To set an aircraft to use this flight mode you use the Speed/Altitude window to set an aircraft (or aircraft waypoint) to use terrain-following mode. Once you set terrain following (and assuming the aircraft is capable of land-cover masking), a new check box will appear in the UI to set 'Land Cover Masking.' This can also be done via Lua (see below.)



Aircraft flying 'land cover masking' will fly much lower than the existing terrain following mode, as low as 10 ft AGL at loiter speed in open terrain with good visibility.

If the scenario has the 'Terrain Effects' feature enabled, an aircraft flying 'land cover masking' will fly just low enough to be inside the land cover when possible. For example if land cover at the aircraft's location is 'urban/built up' with a height of 9m then the aircraft will fly at 8m. This allows the aircraft to receive concealment from the land cover (see sensor changes below) while still exposing any mast-mounted sensors it may have (a good example is OH-58D with its mast-mounted optics).

Land Cover and Sensor Detection Ranges

The effect of land cover on the maximum sensor detection ranges for visual, IR, and radar sensors has been changed. (NOTE: as before, land cover only effects sensor detection range when the 'Terrain Effects' feature is enabled in the scenario settings.)

Previously, only the land cover at the target's location influenced sensor detection range. Now the sensor itself is checked to see if it is 'within land cover' (ie. it clears the skyline or not) and this applies an additional reduction in detection range if the height of the sensor is below the local skyline. In addition, the land cover along the 'line of sight' between the sensor and the target will also reduce sensor detection range whenever the LOS point is below the local skyline.

These changes can result in significantly shorter detection ranges for ground-to-ground sensor detections when there is 'high' land cover in the area (e.g. woods and/or urban terrain.). Very low-flying aircraft (using the 'land cover masking' mode) can also have reduced detection ranges if they are flying below the local skyline (and dodging telephone poles or power lines, probably).

Lua Changes

New Lua support has been added for the features above:

Land Cover Masking Flight Mode

'Land cover masking' mode can also be set via Lua using the Unit wrapper. Set the 'unit.TF' property to 'true' to turn on terrain following and then set the new 'unit.TFType' property to 1 (for land cover masking) or '0' (for traditional terrain following AGL flight.)

Custom Environment Zones

Lua support for custom environment zones has been expanded.

The ScenEdit_AddZone, ScenEditRemoveZone, and ScenEdit_TransformZone functions now support CEZs. CEZs are always added to/removed from the 'nature' side. CEZ values are set via the CEZ wrapper object (see below.)

Lua Side wrapper has a new property and a new method:

Side.Customenvironmentzones read-only property – returns a table of CEZ guid/name/description. Note: only the 'Nature' side will have CEZs normally.

Side:getcustomenvironmentzone(guid or name or description) method – returns the CEZ as a zone wrapper object.

Lua Zone object wrapper:

Zone.landcovertype property – get/set integer land cover type.

Zone.hascustomlandcoverheight read-only property – returns true if CEZ has custom land cover height set.

Zone.landcoverheight property – get/set integer land cover height in meters.

Zone.thermallayerceiling property – get/set thermal layer ceiling in meters.

Zone.thermallayerfloor property – get/set thermal layer floor in meters.

Zone.thermallayerstrength property – get/set thermal layer strength as floating point value.

Zone.convergencezoneinterval property – get/set convergence zone interval as floating-point value.

Zone.weatherprofile property – get/set weather profile as a Lua table with 'temp', 'rainfall', 'undercloud', and 'seastate' table fields with the same values as the weather model.

Other things

* ADDED: Added support for "multi-staged missile" flag. Multi-staged AAW missiles (e.g. SA-5) have a lower burnout weight in proportion to their overall launch weight. This helps them achieve better climb performance post-burnout, but if they loft it also means that they are less able to use their mass to re-energize on their post-loft dive.

- * ADDED: Indication of which AC is the leader in an airgroup
- * TWEAK: Mission's WRA now supports queued dynamic units.
- * TWEAK: Battery-powered torpedoes accelerate much faster than thermal ones
- * ADDED: Added support (incl. damage mechanics) for new "blimp" airship type
- * FIXED: ACs with standoff weapons where flying toward the target after the WPnLaunch
- * FIXED: Importing a unit with extra parking facilities reports overfilled for every extra aircraft
- * FIXED: Special Message Window Sizing
- * FIXED: #14909: Add "Ground Units" to Event Triggers
- * FIXED: [Mission Editor] Deleting all missions in ME was impossible due to retained reference to the last mission selected; this then caused the last deleted mission to "resurrect" upon any interaction (such as assigning a new unit)
- * FIXED: CTD when Clicking on ATO or flightplan editor button in ME with no mission selected
- * FIXED: Exception in Balloon popup was causing a multitude of messages to appear on screen
- * FIXED: [B1307.4] Strike Mission 'Enforced' Flight Size seems to max out at 6
- * FIXED: Early Experience with the New Strike Planner (points 1 & 2)
- * FIXED: #14892: Imported facilities don't maintain heading
- * FIXED: Lua SetTrigger not allowing target filter of side only
- * FIXED: Lua unit course - allow any desired speed up to unit maximum
- * FIXED: #14904: Not all manually allocated missiles firing
- * Includes the new v499 release of the DB3000 and CWDB databases.
DB3000 changelog: [https://drive.google.com/file/d/1XmuwnZ ... sp=sharing](https://drive.google.com/file/d/1XmuwnZ...sp=sharing)
CWDB changelog: [https://drive.google.com/file/d/1Y-vuNJ ... sp=sharing](https://drive.google.com/file/d/1Y-vuNJ...sp=sharing)
- * Includes a teaked & corrected version of the "Showcase #2: Ford Class" DLC scenario. The issue with the squadron selection at the start of the scenario has been fixed.

Build 1307.7 Release Notes (changes from B1307.6)

=====

- * Performance improvements relevant to Showcase: Ford class and the new skyline features.
- * Fixed: Export Inst Issues (Export to file missing Vehicles)

Build 1307.8 Release Notes (changes from B1307.7)

=====

- * ADDED: Added new map layer: OpenTopoMap. This is a map layer optimized for displaying natural topography (see also: <https://opentopomap.org/>) . This can be useful for better evaluating the terrain around a given unit (and thus better understanding visibility, movement and damage effects). This layer is externally-provided, so tile-download reliability and speed are beyond our control.
- * More performance improvements when using the "land-cover effects" scenario feature.
- * TWEAK: [INST]: Add speed and altitude to file
- * TWEAK: Tweaked INST to handle hosted units better
- * TWEAK: [Lua] Made errors clear for interactive use
- * FIXED: [Lua] GetMissions - allow name or guid as input
- * FIXED: [B1307.7] Error in Exception Handler pop-up
- * FIXED: New Ground-Unit Contact being classified in "New Contact" category instead of "New Gound Contact" message window tab

- * FIXED: #14910: Units not Obeying WRA
- * FIXED: [B1315] Exception Error

Build 1307.9 Release Notes

=====

- * The new "Sensors blocked by skyline at sensor location and along the LOS path to the target" feature (introduced in B1307.6) has now been isolated into its own separate "Effects of Terrain Type - ADVANCED" scenario realism option. This option is disabled by default, and should NOT be used unless you know what you are doing. This was done in order to avoid breaking most scenarios where e.g. radars are placed in the middle of forests or urban sprawls without blockage considerations (Recent example: <https://www.matrixgames.com/forums/view...6#p5084166>).
- * FIXED: Severe performance issue when checking if a specific table column exists on the database.

Build 1307.10 Release Notes (changes from B1307.9)

=====

- * ADDED: [Event Editor] New trigger type: "Air/Dock Ops Status Change". This trigger fires whenever the air/dock ops status of the selected unit(s) changes - e.g. an aircraft takes off, a boat returns to its host unit, an aircraft shifts from "being turned around" to "ready and waiting" etc. The UI allows selecting which of the available states will cause the trigger to fire. This was available until now only through scripting but can now also be configured through the EE.
- * FIXED: Error Exception (Command v1.05 - Build 1307.5)
- * FIXED: Destroyed leaders were not passing the group flightplan to the next leader, causing the flightplan to be lost
- * FIXED: [FIXED] Incompatible Flight Plans and Aircraft Armaments - Investigating a Potential Issue
- * FIXED: Exception in Formation editor
- * FIXED: Export of group members not working unless showing Unit View
- * FIXED: [B1307.9] Surface duct effect (sonar) persists at deeper depths (Ref: <https://www.matrixgames.com/forums/view...0&t=393911>). Tweaked this so that the benefit of surface ducting begins (very slightly) at 50m depth and increases in percentage as the sensor and/or the target nears the surface. The maximum benefit (1.5x of nominal direct-path range) happens only when both the sensor and the target are on the surface.
- * FIXED: MDSP #5 Tutorial Oddness
- * FIXED: 1307.5 Mission Editor (Activation Time) crash
- * FIXED: Pre-planned Strike Missions Broken in Build 1307.9
- * FIXED: DB viewer crash when no DB link established
- * FIXED: [Lua][B1307.8] SetTrigger ExitArea not works as expected
- * FIXED: Land Cover at sensor location was applied to IR/Radar even without 'advanced land cover effect' option enabled
- * FIXED: Land Cover Masking checkbox shouldn't show up on waypoint in SpeedAlt if unit isn't capable
- * FIXED: Land Cover Masking check box shouldn't appear on SpeedAlt when unit is not capable
- * FIXED: game crash if SlugTrail_Lifetime is 0
- * FIXED: [B1307.8] Strike Planner Shows Same Target for Multiple Flights
- * FIXED: Various (crash/behaviour) fixes and tweaks for vehicles
- * FIXED: Scripting issue in Showcase - Ford Class scenario.

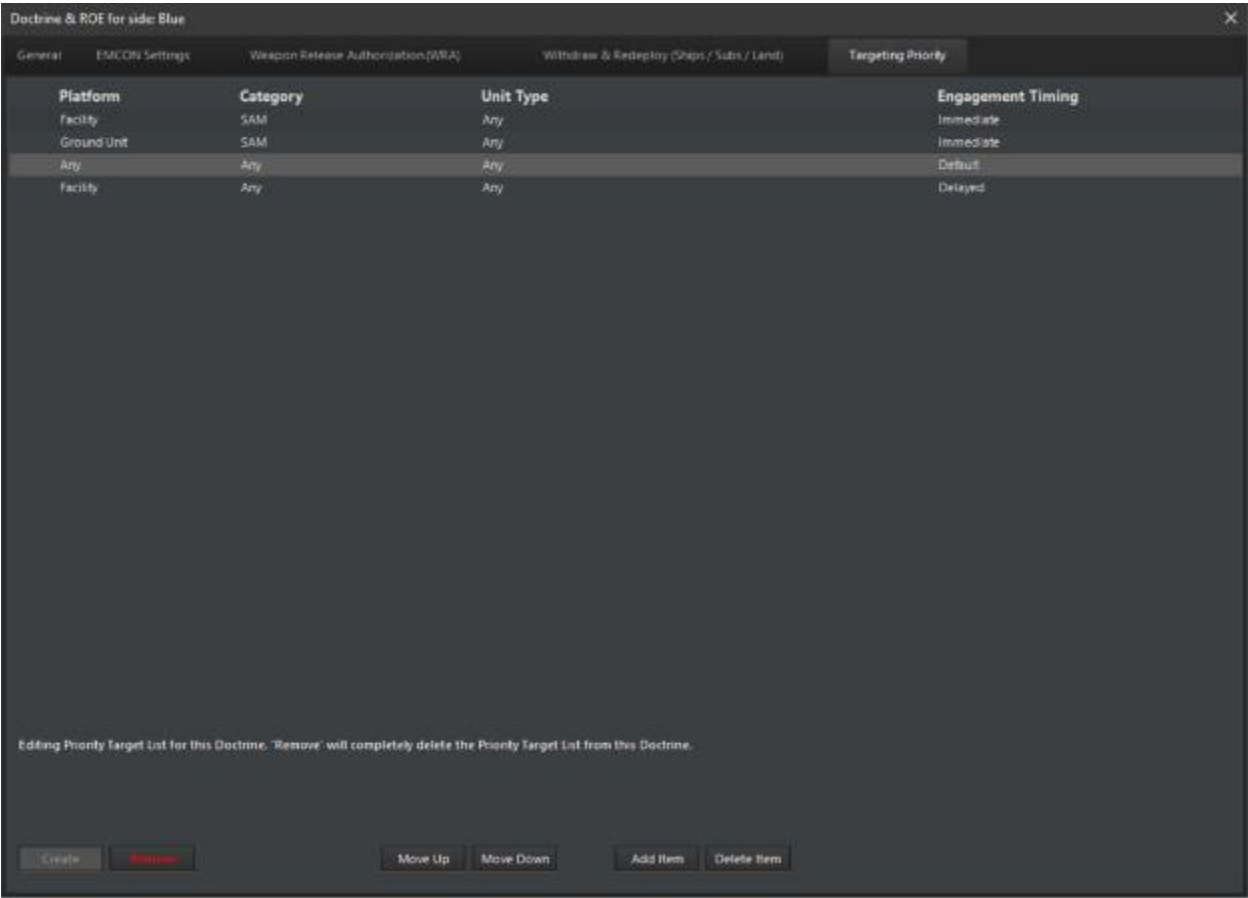
Build 1307.11 Release Notes

=====

- * **NEW MAJOR FEATURE: Targeting Priority** (part of Doctrine/ROE settings):

A new 'Targeting Priority' tab now appears on the Doctrine/ROE window. This tab allows the user to create a Targeting Priority List to control how a unit prioritizes engaging known targets.

Targeting Priority is a component of a Doctrine, and thus follows all the existing conventions for Doctrines in Command – they can be assigned at the unit, group, mission, or side level and will inherit from higher-level Doctrines:



The Targeting Priority List UI will display the Targeting Priority that applies to the Doctrine you’re currently viewing. You can only edit the Targeting Priority List if it is a part of the current Doctrine – i.e. not inherited from a higher-level Doctrine.

Every Targeting Priority List will contain a single ‘default’ priority entry. Entries above the ‘default’ entry in the list will be considered ‘immediate’ priority targets, to be engaged before any other targets. Entries below the default entry in the list are considered ‘delayed’ priority, to be engaged only if no higher priority targets are available. The ‘default’ priority entry will automatically match any target(s) that do not match any other entry in the Targeting Priority List, allowing the player to prioritize (or de-prioritize) individual target types while leaving all other target types to fall into the ‘default’ category.

Create: Allows you to create a Priority Target List for the current Doctrine. This will override any inherited Targeting Priority.

Remove: Removes the Targeting Priority List from the current Doctrine. The Doctrine will automatically inherit from higher level Targeting Priorities, if they exist.

Move Up: Move the selected item up one row in the list.

Move Down: Move the selected item down one row in the list.

Add Item: Add a new item above the currently selected item in the list.

Delete Item: Deletes the currently selected item from the list. NOTE: You cannot delete the default priority item.

Doctrine Targeting Priority applies as one of the final criteria when a unit is selecting a target. Other Doctrine settings (Rules Of Engagement, Weapon Release Authorization, etc.) and general AI targeting considerations (side posture, mission applicability, weapon effectiveness, etc.) are applied **before** the Doctrine Targeting Priority is checked. So for example, if a unit does not have a weapon to engage a given target, or that target is not considered relevant to the unit's current mission, the target will not be considered as a potential target regardless of Targeting Priority.

Each entry in the Targeting Priority List is evaluated in turn against a unit's final potential targets. If an 'immediate' priority entry is matched, the unit will engage the target(s) that match that entry, ignoring lower priority targets. If a 'delayed' priority entry is matched the unit will engage the target(s) that match that entry only if there are no higher priority targets. If there are multiple potential targets remaining after the Targeting Priority is applied the Command AI will select one of those targets based on its general AI and/or mission target selection behavior – typically closest-first, while attempting to spread attackers across multiple targets.

NOTE: In some cases the unit AI will evaluate a nearby hostile unit as an immediate threat and engage it in preference to all other targets. The Priority Targeting List is ignored in this case.

Targeting Priority Doctrine can be changed via Lua using the Doctrine wrapper object. The Doctrine wrapper can be obtained from a unit, group, mission, or side wrapper object via its doctrine property. See 'Lua Changes' below for further details.

Lua additions for Targeting priority feature:

ActiveUnit wrapper: **ActiveUnit.dctrine** read-only property – returns the Doctrine wrapper object for the unit's Doctrine.

Group wrapper: **Group.dctrine** read-only property – returns the Doctrine wrapper object for the group's Doctrine.

Mission wrapper: **Mission.dctrine** read-only property – returns the Doctrine wrapper object for the mission's Doctrine.

Doctrine object wrapper:

* **Doctrine.TargetPriority** read-only property – returns a Lua table with the contents of the doctrine's priority target list, if any. Fields for each item in the table are:

-- unit_type (integer) = ActiveUnitType

-- unit_subtype (integer) = AircraftType, ShipType, SubmarineType, FacilityCategory, WeaponType, MobileGroundUnitCategory, depending on unit_type.

-- Isfixedfacilitysubtype (boolean) = For fixed facility unit_type, 'true' indicates the unit_subtype is a value from FacilityCategory, 'false' indicates the unit_subtype is from MobileGroundUnitCategory.

-- dbid (integer) = the specific DBID.

-- priority (integer) = 0 (delayed), 1 (default), 2 (immediate) engagement priority.

* **Doctrine.deleteTargetPriorityEntry(index)** method – deletes the corresponding item from the priority target list, if it exists. NOTE: index value is 0-based. Returns the list as a Lua table.

* **Doctrine.addTargetPriorityEntry(type, subtype, isfixedfacilitysubtype, dbid, index)** method – all arguments are integers save for 'isfixedfacilitysubtype' which is a boolean value. 'type', 'subtype', 'isfixedfacilitytype', and 'dbid' are as explained for the TargetPriority property above. 'index' is the location in the list where the new entry will be inserted, with the existing entry at that location moving down. NOTE: index value is 0-based.

Other things

* ADDED: [Area/Zone & Refpoint Manager] Interpolation function for zones; Also added Shift+RMB zone interpolation options (1, 2, 3, 4, 9) in the UI

- * ADDED: Added a new Shift+RMB interaction on zones: Highlight all RPs of a zone exclusively
- * ADDED: Shift now allows to move multiple RPs while keeping relative position. This makes it possible to move a selected zone in one go.
- * ADDED: Various features to easily create usable zones
- "Define area" now prompts the user to automatically create a standard zone and requests for a name, expanding the area-centric changes.
- Added a circle shaped "define area" in addition to the rectangle shaped one (shortcut ctrl+o)
- Reorganized the UI for reference points and areas
- * ADDED: Zone selection paradigm for all types of mission areas, triggers, actions, nonav & exclusion zones, and flight plans: <https://i.imgur.com/QJDYIGZ.gif>
- * ADDED: Special kinematic behaviour for hovercrafts (LCAC, for instance), they no longer behave like martians flying saucers capable of hovering above trees.
- * ADDED: Traction system for ground units (also fixes units getting stuck in sloppy area and adds collision bounce)
- * ADDED: Machine information in exception log and current scenario and current database info in written exceptions
- * ADDED: Allow group formation data to persist in INST

- * TWEAK: Updated Cargo load/unload times to handle combinations of towed / carried / personnel cargo
- * TWEAK: Improvements to "Custom Map Layer" window. You can now re-arrange the order of the custom layers, and enable/disable them individually: <https://i.imgur.com/8KaQZIs.png>
- * TWEAK: [Lua] Wrapper unit updates:
 - Allow unit formation to jump to location
 - Added groupLead as a method

- * FIXED: #14942: Group operator not working in ScenEdit_SetUnit Function
- * FIXED: #14919: [LUA] Waypoints Assigned to Groups Aren't Followed
- * FIXED: GMTI & SAR modifiers against mobile aimpoint facilities were not also applied to new-annex ground vehicles
- * FIXED: Fix for aircraft not returning to EMCON properly after aerial refueling
- * FIXED: Typo causing repeated exception trying to determine sonobouy icon
- * FIXED: RP Interpolation for zone was duplicating a RP

Build 1307.12 Release Notes

=====

- * ADDED: Added ability to limit lists to existing units only when adding Target Priority in scen edit mode
- * ADDED: New zone creation "wizard" (from highlighted RPs) in Missions + Ref. Points menu
- * ADDED: 2 new Zone Shift+RMB interactions : RENAME and DELETE (zone/zones+asociated RPs)
- * ADDED: Various features to easily create usable zones)
- * ADDED: Cargo Delivery mission now supports vehicle self-transport
- * ADDED: "Refuel" Title in the Doctrine Form group was missing
- * TWEAK: Wholly revised calculation of drag coefficient for boost-coast missiles, based on the equations on Eugene Fleeman's "Tactical Missile Design" (Example: <https://imgur.com/a/ok0hRwT>)(huge props to Ian for guiding research). For many missiles this results in greater auto-calculated boost-burn times, which in turn results in improved performance at lower altitudes.
- * TWEAK: Made Priority Target Entry form re-sizeable
- * TWEAK: Align ME Home base logic
- * TWEAK: [Lua] UseCustomIntermittentEmissionOnly added as a unit's scope attribute
- * TWEAK: [Lua] ScenEdit_HostUnitToParent
- * FIXED: DB Viewer shows 'On Water' for amphibious vehicles with only one engine alt band
- * FIXED: #14956: Crash occuring when interacting with a destroyed event in the UI
- * FIXED: #14954: Specs fetch in benchmark
- * FIXED: Weapon expenditure count
- * FIXED: CTD when editing or creating nav or exclusion zones when no side is defined

- * FIXED: CTD when attempting to merge units without defined side (such as in the starting menu)
- * FIXED: CTD when selecting quick turn around without defined side (such as in the starting menu)
- * FIXED: CTD when RTB a unit without defined side (such as in the starting menu)
- * FIXED: CTD when opening doctrine form in starting menu
- * FIXED: SQLite exceptions causing missing units when rebuilding a scenario and using an INI file
- * FIXED: Re: Incompatible Flight Plans and Aircraft Armaments - Investigating a Potential Issue
- * FIXED: If a sensor tracking/illuminating a target for fire-control is not able to re-acquire, stop tracking the target even if it is still detecting by another non-tracking sensor
- * FIXED: #14945: Scenario "pausing" every minute
- * FIXED: #14938: Issues with manual Firing allocation of certain weapons
- * FIXED: #14932: Not firing the rounds it should
- * FIXED: #14927: 1307.9 Order of battle Exception
- * FIXED: #14924: [1307.9] Filtering of flight plans is not working well when flightplan is opened from mission editor
- * FIXED: "Unit Destroyed" trigger fires when unit side is changed
- * FIXED: [Lua] Issue with ScenEdit_HostUnitToParent
- * FIXED: AAW missiles with mid-course datalink guidance could be guided even if the guidance provider was tracking the target with a non-AAW-FC-grade sensor (e.g. Mk1 Eyeball)
- * FIXED: AAW missiles with a <5-sec burn time (e.g. SR-AAMs) were not accelerating fast enough in order to reach their maximum speed-at-altitude at burnout
- * FIXED: #14963: 1307.11 Mission Editor Home base selection not showing grouped airbases
- * FIXED: Messages with identical time stamp shown out of sequence in message log UI

Includes fixes and tweaks on the following scenarios, by Brandon:

- * Chains Of War #1 - Blue Dawn
- * Command LIVE - Kuril Sunrise
- * Command Showcase - Queen Elizabeth
- * Choking Halifax
- * Raid On Kismayo

Build 1307.13 Release Notes

=====

- * TWEAK: Tweaks to missile lofting logic (remove arbitrary "ceiling" of loft arc and bias the trajectory angle upwards during boost). Related discussion: <https://www.matrixgames.com/forums/view...0&t=394534>
- * TWEAK: [Lua] Unit set side no longer deletes and adds unit
- * TWEAK: [Lua] Unit manualAltitude allows altitude in FT or M
- * FIXED: Cargo transfer unit wouldn't return to source when destination was a group
- * FIXED: #14993 Too long category name in Targeting Priority WRA
- * FIXED: Hyperlink not working when unit not selected
- * FIXED: [B1307.1] Unit Panel warning message not great
- * FIXED: #14998: Game drops out of high speed to 1x repeatedly then does not resume high speed
- * FIXED: Groups Do Not RAS to Completion Consistently
- * FIXED: TVM-guided weapons were not guiding correctly (affects Patriot/SA-10/20/21 etc.)
- * FIXED: All radar sensors were assumed to be able to generate AAW-FC quality data, even if they are fundamentally imprecise (e.g. low-frequency, OTH etc.)
- * FIXED: Confusing "generation" terminology on agility-based aircraft WRA-type
- * FIXED: Ships and Amphibs assigned to cargo mission were launching without cargo
- * FIXED: Enabled dogleg generation when required by doctrine and if not in BOL

* Includes improved graphics for the depiction of mount/sensor coverage arcs. Example: <https://imgur.com/a/M4cZsm8>

* Includes updated and fixed versions of the following scenarios, by Brandon:

- Falklands #8 - Pebble Island Raid
- Falklands Bonus #1 - The Empire Strikes Back
- Falklands Bonus #2 - Operation Mikado
- Kashmir Fire #13 - Over The Hump

Build 1307.14 Release Notes

=====

* ADDED: #14985: Opacity validation by "enter" key in area & ref point manager

* ADDED: Zone creation by polygon

* TWEAK: Air Ops screen cleanup

* TWEAK: Improved and standardized mission editor date parsing to fix an issue with certain localizations

* TWEAK: #14922: intermittent emission feedback was not hidden when no sensor was active

* TWEAK: Displaying a huge "Drawing Zone Polygon [ESC to cancel]" when in polygon mode (instead of the message in status bar)

* TWEAK: Various improvements on UI barks:

-Automatic duplication detection when queuing a message; avoiding any duplicated bark messages in the queued message list

-Barks now follow anchored units

-Adding a new bark to a unit with already an active bark will automatically queue a message to it instead.

-Queued messages can be customized with all bark's initial customization options

-Fixed an issue where a message next in the sequence would not reattach to the anchored unit

* FIXED: #14244 - Show radar illumination arc when guiding SARH weapons

* **FIXED: Long-burn AAW missiles were not terminal-diving on their target if they were still boosting at the time** (Example: <https://www.matrixgames.com/forums/view...0&t=394652>)

* FIXED: Incorrect calculation of off-board passive sonar sensors (e.g. towed array) detection range modifier

* FIXED: Minimaps Crashing the Game

* FIXED: #14972: It was impossible to group multiple aircraft with mixed DBID

* FIXED: Exclusion of drop target option in Shift+RMB unit interaction , that could lead to issue when selected a contact without a selected active unit

* FIXED: #14929: Weapon are fired above command channel restriction (This behavior is due to a recent change that assumes shared datalinks across group members (e.g. a deployed SAM battery). This behavior has been modified; it is now limited in distance and to ground units that have settled at a given position for at least 2 minutes. So for example once that SAM battery is again on the move it loses the shared-datalink capability.)

* FIXED: #14973: Lead ship not slowing down despite "group lead can slow down" toggled

* FIXED: #15001: Odd pathing of ground units - Given some slack threshold to inertia computation

* FIXED: "Edit Boats" can add the wrong type of unit when number is 'Applied'

* FIXED: Exceptions on Docking Ops form

* FIXED: Mission level WRA does not elaborate on "Inherited"

v1.05.1307.18

=====

• ADDED: WRA shortcut-button in unit status panel

• ADDED: On the "Scenario Migration" window, you can now customize the default WRA-range for AAW missiles when migrating a scenario

• TWEAK: Aligned Editor "Set Unit Property" menu items: Replaced 'Set unit fuel & airborne time' with a new 'Set Unit Properties' which does fuel, air/sea time and alt/depth. Replaced the 'Set time at sea' with call to the above.

• FIXED: #15168: Bug opening formation editor

- FIXED: In some cases the units where requested to fire more weapons then they are carrying, causing fewer weapons to be delivered
- FIXED: #15158: GUI Bug: Mission level EMCON depends on previously selected tab
- FIXED: Do not drop the target if there is a salvo in place but you have to fire infinite weapons
- FIXED: Kinzhal missile flies into space
- FIXED: [Mission Planner] The Mission was looking for a new target if the previous one moved, instead of trying to reacquire it
- FIXED: Strike Plan doesn't work as expected
- FIXED: In some cases ACs where RTB before Winchester or target destruction
- FIXED: Issue in retrieving the proper weapon based on the target type
- FIXED: #15127: [B1307.14] MDSP & WRA Issues
- FIXED: #15048 "Unit is Damaged" Trigger goes off multiple times for the same unit
- FIXED: #15061: Ford Showcase: ScenEdit_SetEMCON doesn't toggle sensors
- FIXED: #14918: Strike Mission WRA not discriminating between Escort and Striker munitions
- FIXED: Un-assigning unit changes condition of a non-operating unit.
- FIXED: Un-assigning unit changes condition of a non-operating unit.
- FIXED: LuaSetTigger() not using correct area RPs for relative bearing
- FIXED: Lua ScenEdit_SetReferencePoint not changing bearing type unless 'highlighted' is present
- Includes all official scenarios rebuilt to DB v499, with their default AAW WRA-range set to 75% of maximum range.

There are some features listed in this document that are only available in the Public beta for the 64-bit/DX11 build. When that becomes the final release, this manual will be updated.

v1.05.1307.19

=====

- Added: #15173: SBR Add Support for Sensor Groups
- TWEAK: Minelaying behavior (Assume if a mine is nearby and not active, and the unit is a ML type, then it laid the mine and it can ignored for now)
- TWEAK: UAV/UCAVs of small size can be catapult-launched and retrieved on platforms without dedicated air facilities
- TWEAK: Restored TO/LA check box on Edit A/c form
- TWEAK: Aligned "Add AC" and "Add boat" window layouts
- FIXED: Submerged contacts never engaged by AI
- FIXED: [1307.18+1328.6] No fire inside no-nav zone (Added a flag to the NoNavZone that designate it as a no fire zone (default is true) so that the user can decide if allowing the unit to fire in the zone or not)
- FIXED: Docking Facilities Issue on DDG 125 (if the first dock facility was unable to host the boat the other ones were not checked)
- FIXED: Deployed mines not appearing in weapons fired export
- FIXED: Missing implementation of "Drifting Mine" behaviour
- FIXED: [1307.18+1328.3] Mission Editor Causing Reproducible Crashes
- FIXED: [Build 1307.18] "Hidden" Thermal layer/databox info broken
- FIXED: Missing illumination arcs on DB viewer
- FIXED: #15188: [B1328.5] Extreme wingman separation
- FIXED: SBR with MCM failing
- FIXED: Strike mission escort flights can not be generated
- FIXED: [1307.18] WRA Different Weapon Allocation
- FIXED: CEZ transparency was not serialized and deserialized
- FIXED: CEZ custom altitude was not able to be set to a value < 0
- FIXED: MPAs and Helis not dropping Sonobouys

- FIXED: #15139: Mission Level WRA GUI: Choices Revert Back to "Various Settings"
- FIXED: [Lua] .Phase in mission wrapper
- FIXED: PCLS broken by incorrectly applied intermittent-EMCON clause
- FIXED: ACs were RTBing without fire (specific case)
- FIXED: CEZ Transparency was not set rendered correctly. Transparency value was not set if changed via text and not via up down arrows
- FIXED: #15172: [B1307.X] SBR Generate Delta Template Errors
- FIXED: crash on scenario reload
- FIXED: Add sub unit a 0 kts (not at creep)
- FIXED: Mk1 eyeball is now operational at periscope depth if no other visual sensor are on platform and scenario is using CWDB
- FIXED: Clicking on "sensor" hyperlink in contact recent detections list does not bring up the DB-viewer as expected if the sensor is in a loadout
- FIXED: Logic flaw in Priority Target List doctrine evaluation may lead to infinite loop (ie. sim freeze)
- FIXED: Weapon "Max Downrange" calculation returning infinity if MaxAltitude_AGL = 0 and MinAltitude_AGL = 0

v1.05.1328.7

- =====
- ADDED: [Lua] Property 'leadSlowsDown' for group wrapper object
 - TWEAK: The supersonic friction increases detection range only for IR, not visual detections
 - TWEAK: AAW weapons get expanded "outer range limit" against HGVs (same as with BMs/RVs and sats)
 - TWEAK: "Game Options" window has border on right and bottom sides as well as left side and is tall enough to show all elements on the Map Display tab.
 - TWEAK: Extra factor added in vs-aero DLZ checks: Target vertical speed. This enables SAMs to have expanded shot opportunities against fast-diving targets (HGVs, high-diver weapons like Kh-22/32 etc.)
 - FIXED: The "Torpedo - ASW Optimized" warhead type was properly evaluated for hull-penetration when detonating
 - FIXED: Torpedo being able to engage other torpedoes even if they are not valid targets
 - FIXED: WRA firing range settings not working after DB upgrade
 - FIXED: HGV/ABM Intercept Fix
 - FIXED: [Build 1307.18] "Hidden" Thermal layer/databox
 - FIXED: Benchmark window shows correct amount of memory on systems with more than 16GB of RAM.
 - FIXED: [1307.18][1328.6] No fire inside no-nav zone. Added a flag to the No-Nav Zone that designates it as a no-fire zone (default is true), so that the user can decide if allowing the unit to fire in the zone or not (<https://www.matrixgames.com/forums/viewtopic.php?f=10230&t=396732>)
 - FIXED: "Unit Status" panel no longer attempts to download missing database unit-images continuously
 - FIXED: Missing implementation of "Drifting Mine" behaviour
 - FIXED: Docking Facilities Issue on DDG 125
 - FIXED: Strike Mission Altitude Bugs
 - FIXED: [1307.18 / 1328.3] Mission Editor Causing Reproducible Crashes
 - FIXED: Scenario Grinds to a Halt
 - FIXED: Missing illumination arcs on DB viewer
 - FIXED: Stamen Terrain and OpenTopo layers render down to zoom level 18.

- FIXED: HGVs with seeker activation point were diving towards the activation point, not the desired target
- FIXED: #15188: [B1328.5] Extreme wingman separation
- FIXED: Radar with no illumination properties was accepted as suitable director for SARH/TSARH weapon
- FIXED: HGV pulls up sharply if it cannot locate its assigned target
- FIXED: Null-reference exception if serializing contact whose actual unit has been nullified
- FIXED: Weapon crashes into ground because of erroneous AGL-altitude calculation
- FIXED: Land cover "skyline height" not displayed correctly when showing altitude in feet
- FIXED: Secondary damage of land units in CEZ
- FIXED: CEZ movement for land units was not possible if CEZ was on land but with bad see state defined
- FIXED: Secondary damage of land units in CEZ
- FIXED: CEZ movement for land units was not possible if CEZ was on land but with bad see state defined
- FIXED: [B1328.2] Flight Leader Won't Engage Hostiles and subsequent
- FIXED: SBR with MCM equipment failing
- FIXED: Ships where unable to navigate in CEZ-defined water areas
- FIXED: Aircraft could be forced in a state of "preparing to launch" causing it to bypass launch check if a condition timer was previously set
- FIXED: Aircrafts should not be allowed to take-off from a facility at an altitude above its max altitude
- FIXED: Ballistic missile would never hit the ground and glide indefinitely
- FIXED: Firing resolution for ballistic missile would cause a CTD (CheckWeaponMinMaxSpeedGate)

v1.05.1328.8

=====

- FIXED: Loitering weapons were not considered a valid unit to issue orders / plot course
- FIXED: Auto-detectable units were not able to be targetted by HARMs or similar weapon because their emission age was too old
- FIXED: NGS buoy (ship), should not get Mk1 Eyeball sensor
- FIXED: Laser designators failed to illuminate for laser-guided weapon
- FIXED: #15210: [Lua] Can not set path to latitude=0 longitude=0 via SetUnit method
- FIXED: #15211: [Lua] Error with ScenEdit_GetSpecialAction(action_info)
- FIXED: #15197: All avgas tanks are empty (also for any unit without propulsion would have emptied tanks)

v1.05.1328.9

=====

- ADDED: Ctrl+P shortcut for polygon zone creation mode
- ADDED: Ability to add fuel records to fixed facilities via scen-editor "Set Fuel" window
- ADDED: [Lua] Support for new cargo types in UpdateUnitCargo
- ADDED: Error window with instructions on how to provide feedback for effective investigation (example: <https://imgur.com/a/ivvMZAs>)
- ADDED: [Lua] SE_UpdateUnit to support adding new fuel source on fixed facilities
- TWEAK: Upon startup, the program attempts to select for use the first AMD or NVIDIA GPU that is available. This _should_ fix the tendency of Windows to assign the sub-par integrated-GPU to Command (because it thinks it's a low-GPU-power desktop application).

- TWEAK: Improved the way the Desktop Window Manager (DWM) uses the rendering pipeline.
- This `_should_` resolve reported "tearing" effects on the map. It also provides a minor performance boost.
- TWEAK: "Stamen Roads & Cities" layer does not display level 13 tiles to prevent map showing what looks like a broken layer with missing tiles (Reason: Stamen server has inconsistent and spotty tile coverage at that level)
- TWEAK: "Stamen Roads & Cities" layer tiles render without tiles missing even when Placenames layer also enabled in View menu
- TWEAK: ROV with fuel source
- TWEAK: Placename labels render with less black fringing around them.
- TWEAK: Map does not freeze if renderer not initialized or in the midst of resizing at the start of a rendering loop.
- TWEAK: Arctic sea ice boundaries display if "borders and coasts" layer enabled.
- TWEAK: Map caption bar displays SOFTWARE RENDERER for GPU if "Show Diagnostics" game option is checked and DX11 initialization fails to initialize the primary adapter.
- FIXED: "No Preferences" Max flight size breaking strike missions
- FIXED: "Land Type Effects - Advanced" option not being saved/restore
- FIXED: Missions involving aircrafts were no longer working
- FIXED: Unable to load some cargo container contents with US units of measure selected
- FIXED: #15195: BMs not reaching terminal point and having broken trajectories when allocating salvo in sim resolutions higher than 1.0 (ie. double-flame mode)
- FIXED: Crank-and-Drag not working?
- FIXED: When loading a scenario, ToT timers in could lead to crash due to decimal's culture (comma-vs-period)
- FIXED: #15202: [1307.19][1328.6] Interception Missions: Max flight size doesn't work in second interception
- FIXED: Loading scenario with taskpool was broken (caused stack-overflow)
- FIXED: #15196: [1307.19][1328.7] Manual Fire against units within No Nav zone
- FIXED: Side-level doctrine priority target list was not being applied
- FIXED: Bypass "Has Local Track On This Contact" check for autodetectable units
- FIXED: Protect against no sensors on sonobouy
- FIXED: #15254: [B1328.8] Extreme wingman separation
- FIXED: Do not fix WP altitude that are already correct
- FIXED: Various weapon quantities converted to "all weapons" setting when firing
- FIXED: #15260: Strike Mission Flight Plans Not Loading or Not Saving Correctly
- FIXED: More refined check for bombing target on land when the IP is oversea
- FIXED: Tankers with buddy stores (e.g. MQ-25) not being refueled from
- FIXED: Mine-clearing mission failing with ROVs
- FIXED: ABM interceptor can have wrong pitch & fail intercept against ascending target
- FIXED: #15248 Unable to Save Game - STX (0x02) char isn't properly handled, causing save crash with some propulsion entries.
- FIXED: #15231: Some SAMs don't fire because they do not activate their FCRs automatically
- FIXED: #15214: BMD Interception issues (wrong intercept point calculated)
- FIXED: Ship pathfinder ignores sea ice
- FIXED: AC were heading north without following the FP
- FIXED: Evaluation if the Take-Off Location moved was never executed, that could lead to a wrong TOT when AC depart from a ship
- FIXED: #15244: Exception occuring on Ref point copy
- FIXED: Pallet weapons were delivering the wrong quantity of ammos when 1 AC was firing

- against multiple targets
- FIXED: #15233: Command freezing due to concurrency issue on non-concurrent dictionary
- FIXED: Missing MCM sensors
- FIXED: #15222: Anti Ballistic Missiles large lateral error performing intercept
- FIXED: ABM intercept point may 'jump' back and forth between two points
- FIXED: #15161: Active GPU appears in map caption bar only if "Show Diagnostics" enabled.
- FIXED: #15227: Custom layers imprint black areas on the map where they are fully transparent
- FIXED: #15221: Anti Ballistic Missiles running in circles after failed interception
- FIXED: [Build 1307.19] Aircraft don't drop unguided bombs: flightplan issue?
- FIXED: Features + Realism "Weather Day/Night" - Not Saving
- FIXED: Lateral error in ABM intercept when future ballistic trajectory was not accurate
- FIXED: BMD problem: If WRA doesn't allow use of the longest ranged weapons of a given type (in this case AAW/BMD weapons) this can prevent the automatic use of shorter-ranged weapons
- FIXED: [Lua] Tool_Range not handling location (0,0)
- FIXED: Exception during contact processing
- FIXED: Autodetectable units cannot be fired at because "the target is not radiating"
- FIXED: Error displaying cargo ops form for ship group

v1.05.1328.10

- NEW FEATURE: Directional high-power microwave (HPM) projector weapons (aka. "EMP guns").
- General background: <https://www.nre.navy.mil/organization/departments/aviation-force-projection-and-integrated-defense/aerospace-science-research-351/directed-energy-weapons-high-power-microwaves>
- Two distinct subtypes of EMP projectors are currently modelled:
- Mechanical antenna emitters like AFRL's THOR/Mjolnir prototype: <https://www.thedrive.com/the-war-zone/thor-microwave-anti-drone-system-downs-swarms-in-test> . These emit a broad "broom sweep" EMP beam to effect single or multiple enemy units concurrently, as in this example: <https://imgur.com/a/KZlm7gg> . This system is present in the DB3000 (Facility #3518)
- AESA antenna emitters like Epirus' Leonidas system (<https://www.army-technology.com/projects/leonidas-high-power-microwave-hpm-system-usa/>), which are able to form and direct multiple tight beams concurrently in order to defeat multiple threats. Single-engagement example: <https://imgur.com/a/zUlk5MA> . This system is present in the DB3000 (Facility #3824)
- ADDED: Shift+Delete shortcut to remove everything in the scenario except the selection and actual units of selected contact
- ADDED: Enable presets on Flight plan Waypoints.
- TWEAK: Reverted to "original" tiles style for Sentinel-2 map layer (related discussion: <https://www.matrixgames.com/forums/viewtopic.php?f=10230&t=396324>)
- TWEAK: Improve the error feedback if, on startup, there is an error during the setup of the initial (blank) scenario
- TWEAK: All recoverable aircraft hve their nominal acceleration values halved compared to the hitherto calculation. This is an interim step towards establishing more faithful values for per-aircraft acceleration. The immediate effect of this is that high-performance aircraft are less able (reduced time window) to outrun/drag-out incoming AAW shots when they detect them.
- TWEAK: The main map window and all secondary windows now use "dark" header bars.
- Tweak: Remove color from sensor in weapon/loadout window
- TWEAK: Cater for loadout weapon records not being present (pod-mounted sensors are collected even if the pod is no longer present)

- TWEAK: Changed column names on Priority Target List form for consistency with other UI
- TWEAK: Lua editor (operation planner loop) now supports resizing and fullscreen mode
- TWEAK: Operation Planner UI now supports resizing and fullscreen mode
- TWEAK: Map rendering pipeline logs issues to exception log if 'Log debug information to file' is checked in game options.
- TWEAK: EMP wave/pulse does not affect visual sensors that are not electronic in nature (e.g. Mk1 Eyeball, binoculars, analog periscopes etc.)
- TWEAK: EMP pulse/wave now also affects engines on aerospace units (e.g. aircraft)
- TWEAK: The "strength" of an EMP pulse/wave dissipates by the square of the distance (not linearly)
- TWEAK: Add build number to the startup line in ExceptionLog file
- TWEAK: Numerous simulation performance improvements
- TWEAK: Aircraft ignoring EMCOM if semi-active guidance weapon is in the air
- TWEAK: The Stamen map layers are no longer available on the map options (Stamen are switching to a paywalled access system: <https://maps.stamen.com/stadia-partnership/>)
- TWEAK: User feedback when flight plan waypoints are automatically adjusted during validation
- TWEAK: Against aerodynamic threats, start targeting and shot-evaluating at an outer-limit value that depends on the target's current mach (provides earlier reaction against hypersonic weapons, if they are detected in time)
- TWEAK: Provide a visual hint to the user on where to find the (very risky) option to let long flights take off with no tankers in place
- TWEAK: Pitch rate on aerodynamic-control weapons reduces with altitude
- TWEAK: [Lua] Additional options for minelaying missions:
 - -
 - MinesLaidInSet: Determines how many mines will be laid in a set
 - -
 - MinesLaidInterval: Determines the spacing interval between mines being laid
 - -
 - MinesLaidMethod: Determines the method used for minelaying (0 = line, 1 = random)
- FIXED: Picking area in area editor would not clear the previous zone
- FIXED: Groups are now taken into account into conditional unit selection
- FIXED: #15330: Group leader having default '0' speed override
- FIXED: #15293: Torpedo with "Datalink Plus Active" guidance (e.g. Mk48 Mod 7) would be stuck into "snake" search pattern
- FIXED: Miscellaneous fixes for pallet weapon logics
- FIXED: Zones cannot be converted to CEZ
- FIXED: Impact logic for AAW missile with proximity fuze (e.g. Stinger-PROX) was not handled correctly
- FIXED: AC slack was too much and it was causing the flights to be on the target too early
- FIXED: #15317: [1328.9] ACs don't take off in Land Strike mission
- FIXED: Typo fix in Game Options window ("Groun[d] Unit Zoom")
- FIXED: #15301: [1328.9] Strike Mission: Max number of flights overwrites Flight Size
- FIXED: #15292: [1328.9] User can't override throttle speed in generated flight plan
- FIXED: Can't remove sensors in loadout
- FIXED: #15312: Desired Speed override conflicting with "Group Lead Allowed To Slow Down" logics
- FIXED: #15296: [1328.9] WRA Gun Allocation not working
- FIXED: Unit with priority target doctrine fires at lower priority targets that come into range
- FIXED: #15305: Radars cannot classify (not identify) aerial targets via NCTR
- FIXED: Custom icon position and centering

- FIXED: Ground unit icon for infantry not being shown
- FIXED: Error in rendering zone with empty description
- FIXED: Icon replacement being drawn off-center during custom icon scaling
- FIXED: [Reference point manager] desktop color was applied to some buttons
- FIXED: [Lua] Standard Zone missing from Lua SE_RemoveZone()
- FIXED: Group leader designation (based on flight plan) should consider flight plan with at least 1 waypoint in it only
- FIXED: Erratic navigation of aircraft on patrol
- FIXED: Strike Mission GUI Bug (Confounded Buttons on Target Page)
- FIXED: RTB (Mission Over) Bingo/Winchester (Missions that generated a flight plan as result of a trigger were not adding the target to mission-specific targets)
- FIXED: Some ACs were unable to understand that they have reached a WP because they were flying in circle with a radius too big
- FIXED: Unit with a comm link with zero max channels available (e.g. Link16 in v501 DBs) could not establish a comm link
- FIXED: #15131 : Barks were no longer working in most cases
- FIXED: #15335: Pallet weapons not working --- Extended pitch rate calculation to take air density into account in "MaxAltitude = 0" cases
- FIXED: Incorrect logic checking for daisy-chain comms could prevent CEC connection
- FIXED: Guided projectile rounds were skipping the "PH reduction at mid/long range because of energy reduction" when engaging aerospace targets
- FIXED: Prevented crash for warhead circular references (e.g. Scud C with submunition warhead) and added silent "DB exception" when this error is spotted
- FIXED: Weapons pallet needs to be able to parachute-loiter even when at full stop
- FIXED: ACs were RTBing just after launch
- FIXED: Speed & Altitude setting disappearing when editing flight plan waypoint
- FIXED: #15300: Another case where parent specific weapon data link needed to be checked
- FIXED: "Manual local control" modifier accidentally also applied to A2A gunnery (leads to unrealistically low PH)
- FIXED: #15327 - [B1328.9] Time stuck
- FIXED: Lofting AAMs/SAMs lose too much energy during terminal dive because they fly straight to the target

Includes updated versions of all official scenarios:

- All scenarios updated to DBv501.
- Changed WRA-range for AAW weapons to 50%.
- Update for the "Basic Air Ops" tutorials to take into account all of the recent mechanic and GUI changes.
- Fixes for "Raid on Kismayo", "Stand Up", and "Surface Tutorial 1.3".
- Added support for Command Showcase - Operation Desert Falcon

Includes the new v502 DB3000 and CWDB databases:

DB3000 v502 changelog:

https://drive.google.com/file/d/19VIGrRxC453noab_vDnSZggdef-5p9NF/view?usp=sharing

CWDB v502 changelog: <https://drive.google.com/file/d/19VBm2pSFZhLkjZBtFGKkjiQXLFZ8z3G7/view?usp=sharing>

v1.05.1328.11

=====

- NEW FEATURE: Counter-RAM mechanics (currently only for unguided rockets). General background: https://en.wikipedia.org/wiki/Counter_rocket,_artillery,_and_mortar

- 0
- Unguided rockets are now no longer auto-detectable, instead they need to be explicitly detected by sensors. Rockets above 100mm in caliber can be detected by general air-search sensors if the physics are right; smaller-caliber rockets can be detected only by CRAM-specialized sensors such as counter-battery radars (e.g. TPQ-36/37, Zoopark-1/2, ARTHUR, COBRA etc.). They can also be engaged by weapons possessing a CRAM capability, such as the Centurion CIWS or Iron Dome.
- 0
- If a counter-battery optimized sensor detects one or more rockets, it is able to backtrack their trajectory and generate a rough-location contact for the unit that fired them. This allows modelling real-life counter-battery hunting tactics.
- ADDED: [Lua] Added actual roll and pitch to unit wrapper
- ADDED: [Lua] Add the name of the causing event/trigger/etc to generated Lua-error messages
- Numerous performance tweaks on map/UI and simulation engine
- TWEAK: Restored the "Placenames" map layer, using a subset of the Stamen Labels layer (cached on MG servers, not Stamen's).
- TWEAK: Ballistic missile and RV contacts are marked as "FIREBALL" (instead of MISSILE)
- TWEAK: [Lua] Enhanced CallAdvancedDialog to handle HTML as input and return Lua table of values
- TWEAK: Tweak on aircraft acceleration physics: The accel rate drops due to exponentially-rising drag as approaching the maximum speed for given alt & throttle (practical effect: planes accelerate really fast from an initial low speed, but take forever to reach the maximum speed).
- TWEAK: In ABM DLZ checks, consider the interceptor's ceiling when attempting to calculate the seeker-activation point
- FIXED: #15079: [B1307.2] ToT not working for single unit firing on multiple targets
- FIXED: #15372: TALDs not being rendered on map
- FIXED: #15334: [B1328.10] Weapon allocations by default?
- Fixed: Torpedo contact impact crashing for undefined primary target
- Fixed: Crash when fetching time of day
- FIXED: #15359: Issue with text measurement on info bar
- FIXED: WPN MAX quantity onboard was miscalculated in some occasion + in some cases the unit was focused on fire against a low priority target
- FIXED: AAW range-ring flickering
- FIXED: ACs not dropping their bombs on the first pass
- FIXED: ACs not dropping their bombs due to altitude
- FIXED: AEGIS ships not auto-firing SAMs
- FIXED: #15365: [B1328.11] SA-21 blocking issue
- Fixed: Wrong behaviour of ground unit slider and Icon zoom not being saved and shown in interface
- FIXED: Climb Time was not calculated correctly for HoldStart and HoldEnd WP 1/X
- Fixed: DLZ failure for ABM intercept
- FIXED: #15314: [1328.9] Intercept Mission
- FIXED: #15338: [CPE 2.2.5] [1328.10] Duplicate row when selecting "preset" in WRA Range
- FIXED: #15355: [1328.10] Copy&Paste a decoy unit add DECOY word every time.
- Fixed: #15357: S-400 Peters out vs. target at medium range
- FIXED: (1328.10a) Naval unit not auto engaging ground mobile
- FIXED: #15361: [1328.10] Priority in land ship weapons
- FIXED: MDSP issues regarding TOTs and Wpn Altitude
- FIXED: Cargo Package not in Pool

- FIXED: [1328.10a] Repeatable Exception Error During Bomb Impacts
- FIXED: [1328.10a] Very Straightforward Strike Mission Errors and not meeting TOT
- FIXED: #15353: [1328.10] AA-12 Tailchase
- FIXED: Allow Ferry missions to use flightplan only in "One Way" Setting
- FIXED: Aircraft Not Following Flight Plan
- FIXED: No MDSP message if the correction if the altitude function is not called from the UI
- FIXED: Logic flaw in determining units stacked under the map-click point
- FIXED: When firing to an incoming weapon if the weapon is within the min range (plus some slack) drop it + Message feedback for unit not firing
- FIXED: Low-DP but nevertheless "valid" (in terms of armor penetration) hits were not destroying ground-vehicle units (this can happen frequently with e.g. cluster munitions).
- FIXED: the Turning point Ingress just before weapon release has to follow a different default altitude, the "strike" one and not the "cruise" one
- FIXED: Taking into account mission template when generating or fixing flightplan altitude

v1.05.1328.12

- ADDED: New map display option (Game Options window): Draw black outlines on icons & text to make them easier to stand out against background map (ON by default; disable if it is slowing down rendering)
- ADDED: [Lua] Added target property to unit wrapper
- TWEAK: Map appears at slightly higher resolution on average by loading higher zoom levels at higher altitudes.
- TWEAK: Updated built-in callsigns list
- FIXED: Crash issue on startup
- FIXED: #15340: [Lua] Sensor mounts were not being attached to parent platforms when added by Lua - SE_UpdateUnit; this led to incorrect illumination-arc calculation
- FIXED: #15335: [B1328.10] Palletized munitions with Nan altitude and other issues
- FIXED: [B1238.11] Rapid Dragon weapons pallets transcending altitude, accelerating infinitely, and migrating to North Pole
- FIXED: [Lua] Exception error in weapon wrapper
- FIXED: [B1328.11] Strikers and Escorts Not Working Together
- FIXED: Lofted ASW weapon (SS-N-15) was considering the target altitude (underwater) instead of the desired impact altitude (water surface) for the lofted trajectory calculation
- FIXED: [B1328.11] Aircraft egressing off flight plan to the east and west for some distance
- FIXED: #15386: [B1328.11] Aircraft selected not transfer to Target list on air strike mission
- FIXED: Missiles where not splashing on water
- PARTIALLY FIXED: #15375: [B1328.11] DF-26 No altitude and weird behavior
- FIXED: #15304: [1328.9] Air Intercept, low altitude in transit
- FIXED: Escorts flying in circle ignoring the escorted unit
- FIXED: Setting sub/ship throttle causes sonar signature to increase even if vessel doesn't move
- FIXED: #15377: Speed override causes Sub to keep moving after course is cleared
- Numerous general fixes
- Numerous sim-performance tweaks

Includes the new v503 release of the DB3000 and CWDB databases.

o DB3000 v503 additions:

<https://drive.google.com/file/d/1CfqVLnI4pR1oSHbD4hUOYPzPLzgyx7Ou/view>

o CWDB v503 additions:

v1.05.1328.13

- NEW FEATURE: Water-surface facilities (supported only on v504+ DBs). Such facilities are placed on the water surface (not on the sea bottom) and can be attacked with both land-attack and anti-ship weapons. Examples of such facilities include bridges (both single-unit and multi-segment), offshore platforms, piers etc.
- ADDED: Unit Status panel (right column) shows primary target name for selected unit
- ADDED: Display a red X-crosshairs symbol and ETA on the calculated interception point when a weapon is selected
- ADDED: Added 'Unpack all containers at destination' option for Cargo Transfer Mission
- ADDED: Added aircraft emergency diversion to new host base when fuel is too low
- ADDED: [Lua] Contact wrapper - markedAsDecoy property
- ADDED: [Lua] Add time on target and takeoff time to mission wrapper
- TWEAK: Significant performance improvement on game startup and scenario loading time
- TWEAK: Fixes & enhancements on mine-laying mission
- TWEAK: Losses & expenditures: Don't attempt to show stores with an empty name
- TWEAK: Map appears at slightly higher resolution at lower altitude, reducing OpenTopoMap pixellation
- TWEAK: OpenTopoMap layer does not stop rendering tiles at very low altitude. Virtual Earth layer renders tiles at higher resolution at very low altitude.
- TWEAK: When bubbling up message-box messages from sim-core to UI, check if the originator side is the current side on the UI; if not then don't show them
- TWEAK: Nuclear detonations begin appearing immediately upon detonation instead of after a slight delay and their circles expand with internal fill that fades out over time.
- TWEAK: Event balloons render properly again with outlines and elliptical background fill around them.
- TWEAK: Various refinements on the "throttle down to avoid wierd things" logics active during double-flame time accel
- TWEAK: List Packages in order by name
- FIXED: #15458: Various aircrafts don't have a correct WRA Classification
- FIXED: #15447 : Ballistic missiles erratic trajectories
- FIXED: Comms-jammer was jamming low-frequency radar
- FIXED: Bark for "weapon hold" status and "switching to opportunity targets" were visible even if the unit was not from the player side
- FIXED: Cached datatables access issue causing DB-Viever and "Add Unit" dialog to not behave correctly
- FIXED: Decoy icon not being shown for decoy UAVs
- FIXED: [1328.12] OEMC jamming other bands
- FIXED: #15468: No Navigation Zones Prevent Flights from Launching
- FIXED: Missiles were handled as air contacts and so they where not always targeted + there was an hardcoded limitation of 10NM when identifying an incoming weapon as a threat.
- FIXED: #15469: [1328.12] BM detonating into the ground
- FIXED: #15250: Port/dock 'blue area' (pier lane) not working properly
- FIXED: Keep plotted path on map when WP selected
- FIXED: [DX11 renderer] Pier-lanes not being rendered
- FIXED: #15462: [1328.13] Units on inactive missions don't perform manual attack

- FIXED: #15407: [1328.12] SAMs not working against Guided Weapons and BM
- FIXED: #15466: [1328.12] HAAWC from Osprey not seen
- FIXED: Non-Functional Air-to-Air Missiles in Helicopters, CMO v1.06 Build 1328.12
- FIXED: Suicide craft not auto-attacking water-surface facilities, also manual suicide attacks do not generate damage
- FIXED: Various fixes on submarine damage from shaped-charge torpedoes:
 - The secondary-explosion damage of such torpedoes was not affecting the target sub
 - The explosion standoff was too small for all types of ASW torpedo impacts (generated too high shock factor)
 - Fire/flood effects are now cumulative (e.g. major + major --> step up to "severe")
- FIXED: "ASW optimized" warheads not producing secondary blast effect
- FIXED: ASW-optimized torpedo warheads were not getting their designed penetration & flooding bonus
- FIXED: Unit assigned to cargo mission waits for pending cargo when it won't be able to load it
- FIXED: #15396: (1328.11) Illumination and targeting vectors not appearing properly
- FIXED: Weapon losses not recorded
- FIXED: Unit icons "parallaxing" against map layer at extremely close zoom
- FIXED: QuickSink-fused weapon was impacting non-ship target and the impact logics had not accounted for this case (fallback to normal impact)
- FIXED: Weapon ETA calculator did not consider carried-dispenser weapons (JP233, MW-1 etc.)
- FIXED: in some cases Aircrafts where circling around WP 6
- FIXED: #15310: CEZ Terrain Pathfinding Bug?
- FIXED: Explosion crater only damaging the first hit unit
- FIXED: #15253: [B1328.8] 64-bit slowdown when launching mass amount of bombs
- FIXED: Calculation of ABM intercept point did not take into account max weapon altitude
- FIXED: Tanker-to-tanker aerial refueling issue
- FIXED: #15440: Diesel/electric submarine extremely loud if electric engine is destroyed
- FIXED: #15414: [B1328.12] Aircraft not following flight plan speed
- FIXED: #15077: The DISPLAY of Flight Plan Paths seems off
- FIXED: #15402: [1328.11] Prosecution Area not working
- FIXED: (Double-flame mode rudder behaviour) Wobbling ships, boats and submarines
- FIXED: Broken ESM for negative altitude
- FIXED: (Double flame mode) Some weapons type overshooting targets
- FIXED: #15415: [B1328.12] Aircraft not firing all their standoff weapons
- FIXED: #15127: [B1307.14] MDSP & WRA Issues
- FIXED: Some ACs where not RTB
- FIXED: #14926: [1307.9] ATO does not show all packages under a task pool when the pool is selected
- FIXED: Units with a phased-array radar boresighted well away from the unit's heading (e.g. SLAR) incorrectly suffered the "phased array degradation due to off-boresight angle"
- Fixed/Tweaked: Effectively decayed satellite (moving along a decayed orbit) will destroy it
- FIXED: #15431: AOU miscalculates for very high altitudes and sensor range (e.g. GEO SIGINT satellites)
- FIXED: #14925: [1307.9] Sorting ATO by mission/package greys out all buttons except "Create flight"
- FIXED: #15374: [1328.11] Can't assign a ship from a group to a Strike mission
- FIXED: Group leash transparency
- FIXED: 0014882: [B1303] Extreme wingman separation
- FIXED: Leftover UI-only code (from the pre-CMO days) stating that HGVs & D-EMP weapons

- could only be used in CPE
- FIXED: ABM interceptor activation point too low vs. ascending target
- FIXED: #15430: [1328.12] Command Blocked with Error
- FIXED: UAVs visual/icon classification fix
- FIXED: #15418: [1328.12] DB viewer exception Log
- FIXED: #15413: [B1328.12] Aircraft taking off too soon
- FIXED: Issue with the new HPM weapon type that doesn't let you attack an UAV if it's a weapon (ie. loitering munition) but if it's an aircraft you can attack it
- FIXED: #15410: [B1328.12] Aircraft flying off into space (point 1)
- FIXED: #14923: [1307.9] Cloning missions with generated flightplans messes up some references in ATO
- FIXED: #15406: Parabolic trajectory on guided projectiles wasn't taking into account altitude delta
- FIXED: [Operation Planner] UI is more responsive for H and L initial mission dropdown when adding a mission concurrently
- Numerous general fixes
- Numerous sim-performance tweaks

Includes the new v504 release of the DB3000 and CWDB databases.

o DB3000 v504 additions:

<https://drive.google.com/file/d/1InvoyJJJnRGfCpDjgU9cNOAxMNCCw1P9/view?usp=sharing>

o CWDB v504 additions:

<https://drive.google.com/file/d/1Ip8AM2y1EZbQgQusUC6BFPzIKwxU6Cgt/view?usp=sharing>

v1.05.1328.14

=====

- New weapon type: Laser dazzlers. (Background: [https://en.wikipedia.org/wiki/Dazzler_\(weapon\)](https://en.wikipedia.org/wiki/Dazzler_(weapon)))
- Dazzlers differ from high-energy lasers as they have a deliberately wider, more dispersed (and less powerful) beam pattern. This reduces the destructive effect but makes it easier to hit their intended target. They are used primarily to damage/destroy visual/EO/IR sensors, on manned as well as unmanned platforms and weapons. When used against manned aircraft, they can cause temporary blindness and even lead to the aircraft crashing if at low altitude.
- ADDED: #15453: Add ability to place reference points relative to any unit or any reference point
- ADDED: [Lua] Show if comms being jammed on unit.components wrapper
- TWEAK: #15475: [B1328.13] Weapons in final phase always redirected to target
- TWEAK: Rudder turn speed and overswing prevention logic
- TWEAK: Show if comms jammed on Comms window
- TWEAK: Setting "WCS free" now explicitly informs that neutral targets are excluded from engagements
- FIXED: [B1328.13] Patrol & prosecution zone issues
- FIXED: ACs where blocked in refuel loop
- FIXED: #15485: [Lua] TimeToReady inside event not working
- FIXED: Do not perform expensive "Terrain type effects - Advanced" checks for satellites
- FIXED: #15502: [B1328.13] Aircrafts RTB if WP 2 it's close to WP #8
- FIXED: Weapon ETA cache was cleaned every second + Weapon ETA recalc logic
- FIXED: SM-6 Blk IB (fired against ground target) petered out way short of max range
- FIXED: Cater for BOL-attack when checking for water-surface targets
- FIXED: [Lua] Restore visibility to :methods (fixes several reported issues in B1328.13)

- FIXED: #15488: [B1328.13] RPs relative to a unit not being saved
- FIXED: #15486: [B1328.13] Dipping sonar not detected by subs
- FIXED: #15445: [B1328.13] Wrong position of contacts in scenario
- FIXED: RPs relative to units where not working correctly
- Added support for the "Showcase #4 - Icebreakers" scenario.

Includes updated versions of all official scenarios. All scenarios were updated to the v504 DBs, in addition there have been fixes & tweaks to the following scenarios:

- o Falklands - Cry Havoc Bluff Cove: Added Saint Julian Air Base
- o Live - No BREXIT, no Problem!: Fixed USS Missouri Doctrine, should no longer RTB at scenario start
- o Live - You Brexit - You Fix It!: Fixed USS Missouri Doctrine, should no longer RTB at scenario start. Updated Russian AAW loadouts and Swapped German NH90 DBIDs
- o Live - Pole Positions: Fixed US submarines not being assigned to missions when playing as Russia
- o Standalone - South China Sea Clash: Swapped USA LCS DBIDs, CCG WHEC DBID, and Updated UnitDestroyed events

v1.05.1328.18

=====

SIGNIFICANT MODEL TWEAK: Velocity filters for air & space-search radars.

- o Until now, radars with an air-search capability and those with space-search (ie. ASAT & ABM) capability were separated in their target detection capacity by altitude and target type; ie. air-search radars could detect aerodynamic targets within the atmosphere, whereas space-search radars could detect BMs and RVs outside the atmosphere. While broadly accurate, this abstraction left a lot of unrealistic cases: For example a fighter radar could detect an RV in its terminal phase, or endoatmospheric guided weapons such as hypersonic missiles where not detected by BMD radars.
- o IRL, radars use velocity filters (in addition to altitude ceilings) to specialize in their search task and minimize false contacts etc. Command now correctly replicates this behavior, with some assumptions for the minimum and maximum velocity values:
 - Air-only search radars may detect any target from 0 knots (hover) up to Mach 5
 - Space-only search radars (ie. BMD/ASAT) can detect any target from Mach 2 up to "unlimited" velocity
 - Combined air & space-search capable radars use the full velocity spectrum
- o This addition allows more realistic representation of the capabilities and limitations of each type of radar, against the full gamut of endo- and exo-atmospheric targets.
- ADDED: Feedback for "Unit Add" when an airbase cannot host a unit
- ADDED: Autodetectable feedbacks: Added a warning when adding an autodetectable target to Mission target list and added a specific bark message when an engagement is failing because the target is autodetectable but is not emitting (e.g. SEAD strike on fixed but silent radar)
- ADDED: [Lua] New method for creating a new blank scenario: Tool_BuildBlankScenario()
- Includes revised tutorial scenarios by Brandon Johnson.
- TWEAK: ABM-optimized radars can scan all the way up to vertical zenith (instead of 80 degrees)
- TWEAK: Numerous tweaks, fixes & improvements on BMD engagement logics. Not covering all known issues but should resolve most egregious cases (famous last words).
- TWEAK: Error message for unhostable unit behind a check for "Side is Human controlled"
- TWEAK: Using a modified random-number-generator (RNG) to ward off edge cases like constantly returning 0 if hammered from multiple threads
- Tweak: DB-viewer: if re-using window, act on highlighted item if passed

- Tweak: DB-viewer: Re-enable "scroll to" for sidebar links
- TWEAK: [QOL] If you open the side- or mission-level WRA with a contact selected, the contact's relevant WRA category is highlighted (yellow)
- TWEAK: Platforms with a sonar (if selected) will now always render the direct-path range, even if CZs are not possible in their location
- TWEAK: When a contact that is a potential SEAD target (e.g. an enemy radar facility) is detected radiating, its WRA target-type is presented as "[Normal WRA type, e.g. building or mobile vehicle] + Radar". This indicates to the user this contact's "dual" WRA nature. When auto-firing on this target with a SEAD-oriented weapon, the "Radar" WRA-target-type logics (salvo size, firing range, number of shooters etc.) may take precedence over the contact's normal WRA-type logics. When auto-firing a weapon without dedicated "Radar" WRA-type on this contact, the contact's "normal" WRA-type takes precedence instead.
- TWEAK: Proportional navigation for missile terminal guidance has been temporarily disabled (replaced by intercept-calc guidance), as it contains a critical logic flaw. This will be revisited later.
- FIXED: The "if the weapon doesn't have any functioning seekers, it's going to miss" clause was overlooking cases where a weapon is entirely remote-controlled with no onboard guidance (e.g. TVM-guided weapons like Patriot)
- FIXED: #15563: [1328.14] Patriot guidance issues
- FIXED: #15506: [UI] Scen Editor Chapter and Hotkey Window with unnecessary scrollbar
- FIXED: #15561: [1328.14] Refueling with flightplans
- FIXED: Units will not drop munitions on (mobile) ground targets
- FIXED: "Destroyed" event not registering weapon destroyed by another weapon correctly
- FIXED: Edit Cargo form didn't size-check when color coding cargo items in DB view
- FIXED: #15533 : Rate of Fire was no longer working
- FIXED: #15538: [1328.14] Unit engage offensive with no weapons to attack
- FIXED: ABM off course due to terminal point mis-calculation
- FIXED: Selecting a ballistic unit/contact would force an altitude sanity check bringing its altitude to 0 --- this silently broke any selected unit and also broke released weapon (RV) which were released at 0m alt
- FIXED: Intermittent emcon could sometimes initialise with "Wake when detecting threat" on when loading a scenario
- FIXED: #15520: Broken interception of HGV --- Command would freeze
- FIXED: #15535: [1328.14] UI maintains weapon target on side panel
- FIXED: #15530: WRA Broken
- FIXED: #15529: Mission Editor Exceptions
- FIXED: #15526: Aircraft divert to base with no reason
- FIXED: #15519: [1328.13] Active Sonar detect sub, but passive sonar from sub unable to hear Active Sonar
- FIXED: Cargo Package Inside and Outside Task Pool Simultaneously
- FIXED: [1328.12] Can't arm second Caiman
- FIXED: #15367: (1328.10a) Aircraft not being reported ready in message log
- FIXED: #14923: [1307.9] Cloning missions with generated flightplans messes up some references in ATO
- FIXED: #15518 Cargo UI Window Wrong Label Anchor
- FIXED: AC leader can get blocked in circle flight
- FIXED: #15444: [B1328.13] Empty Slots create fake planes
- FIXED: #15541: [1328.14] Command guided missiles overflow target
- FIXED: #15579: [1328.14] WRA Self Defense doesn't inherit from superior echelon
- FIXED: #15574: [1328.14] Max number of flights disregarded
- FIXED: [1328.14] Game Slow-Down due to Missiles (caused by setting missile WRA-range to "NEZ")
- FIXED: #15563: TVM-guided weapons with no active sensor would never impact

- FIXED: Repeated exception checking if mount is busy when it has never fired
- FIXED: Very slow DLZ calculation if weapon WRA-range set to "NEZ"
- FIXED: #15592: [B1328.13] Escorts circle if strike aircraft destroyed
- FIXED: #15537: [1328.14] Support Mission Refuel issue
- FIXED: Command-guided missile not maneuvering correctly to intercept point
- FIXED: Unit was not determining correctly that it had a local track on a given contact. This led to the "Firing unit must obtain (from itself or another CEC-enabled platform) a high-quality track on the target before firing" block-reason being given in circumstances where it should not.
- FIXED: #15576: [1328.14] RTB and Refueling Status
- FIXED: #15541: [B1328.14] Command guided missiles overflow target
- FIXED: #15661: F1 auto-attack not working properly
- FIXED: #15665: NEZ DLZ Cache generation was broken
- FIXED: Notification_Barks is now flushed when releasing references
- FIXED: #15660: THAAD not engaging targets --- Issue when targeting HGVs
- FIXED: #15520: Broken interception of HGV --- Command would freeze
- FIXED: Unhandled UI exception when StandardFormations.txt can't be opened
- FIXED: When logging exceptions, custom exception data were not written out
- FIXED: Dark Blue text on Dark Grey background on DB-viewer (difficult to read)
- FIXED: EMCON not being properly shown on interface and set to units
- FIXED: AIM-9X-Blk2 was lofting at unreasonably high trajectory at a high-altitude shot
- FIXED: #15593: [B1328.13/CPE 2.3.1] Phantom flights error
- FIXED: EMCON not being applied properly to UUVs, ROVs and all MCM platforms
- FIXED: #15578: [1328.14] Flightplans not seen
- FIXED: Non-aggregate exceptions were not logged correctly
- FIXED: #15659 & #15538: Units were dropping from engaged offensive as soon as they were out of current weapon range (or avoided engaging at all)
- FIXED: #15668: Aircraft ignoring hold waypoint times during MDSP strike missions
- FIXED: uncaught exception when deleting a unit in scen edit mode
- FIXED: #15675: BOL order not working
- FIXED: #15669: Add more precision to fuel consumption
- FIXED: Crash when adding selected targets to just created mission if no other missions were present in the treeview
- FIXED: #15594: [B1328.13/CPE 2.3.1] Standoff munitions not arriving at ToT
- FIXED: In some cases Plotted Course where not cleared from the AC during flightplan generation
- FIXED: #15680: Exception-Fuse Error
- FIXED: Sim freeze on weapon DLZ calculation
- FIXED: ACs were cycling if the target of a strike FP was destroyed
- Fixed: #15362: [1328.10a] No impact resolution for inertial weapons at flame speed
- FIXED: #15402: [1328.11] Prosecution Area not working
- FIXED: Ships were moving to a standoff range for the longest weapon without considering Doctrine limitations
- Fixed: Datalinked weapons not receiving midcourse guidance updates
- ADDED: Unit (temporarily) disobeys EMCON to properly guide datalinked or semiactive weapons in the air
- FIXED: #15690: [2.3.1] PrSM with 500lb WH dispensing submunitions (cluster detonation over water was actually happening underwater)
- TWEAK: Air-search radars can also detect hovering air targets
- RE-FIXED: #15519: [1328.13] Active Sonar detect sub, but passive sonar from sub unable to hear Active Sonar
- FIXED: CAP was engaging outside patrol areaa
- FIXED: [1328.17] Air Intercept Mission - Interceptors Stuck on Loiter
- FIXED: Plotted course is ignored if the AC is RTB in Winchester state

- RE-FIXED: #15406: (1328.12) Unguided unitary rockets disappearing before reaching target
- FIXED: Pop-up message - "hold" popup messages were not following "Behaviour Bark" options like other barks
- FIXED: "Investigates unknown contacts" flags were used against targets with different stance
- FIXED: Units were ignoring contacts outside patrol area even when instructed to investigate
- FIXED: Plotted course continuously regenerated in CAP missions
- FIXED: #15734: (1328.18) Aircraft on CAP investigate hostile contacts outside prosecution area and do not engage
- FIXED: #15732: (1328.18) Unit and mission waypoints getting deleted before unit reaches them
- FIXED: ACs went back at FP instead of landing
- FIXED: In some cases MDSP Errors were not initialized before being accessed
- FIXED: Sensors that are kept active to aid in datalinked weapon guidance are limited to the ones that can provide an AAW-FC-grade track
- FIXED: #15715: [1328.17] Waypoints skipped when running the sim below flame speed

Includes the new v505 release of the DB3000 and CWDB databases.

o DB3000 v505 additions: https://warfaresims.slitherine.com/DB3K_505_Changelog.txt

o CWDB v505 additions: https://warfaresims.slitherine.com/CWDB_505_Changelog.txt

Beta build 1405 CIV (Steam only)

We have just put out a Public Open Beta on Steam. Head into the properties section and opt into the beta branch, it will become available to download.

This update is the culmination of a months-long endeavour to consolidate our development branches on a stable, solid release and address the various significant issues reported earlier this year on the B1328.xx series. We have received very positive feedback on the result during internal testing, and we hope this also applies to our wider public userbase.

Because this is a release focused on quality improvement and issue fixes, there are only a few new additions. The most significant are:

* A revised HTML-rendering backend, using Microsoft's new WebView2 rendering engine. You should notice improved HTML rendering on the DB-viewer and other UI elements that use HTML sources. This also allows richer custom UI elements at the beginning of a scenario (e.g. to define custom options for the scenario, difficulty settings etc.)

* Indicators for aircraft contrails and ship/submarine wakes can be displayed on the map, under a unit's datablock. This makes it easier for the player to be more aware that his units may be visually detected at significantly longer ranges because of these factors. These indicators can be disabled on the "Game Options" window (similar to noise & cavitation indicators) if they clutter the map unacceptably.

* The new update also includes the long-awaited new releases of the DB3000 & Cold-War DB databases (yes, the AIM-174B is included in v507).

DB3000 v506 changelog: <https://drive.google.com/file/d/1n0UWMU...sp=sharing>

CWDB v506 changelog: <https://drive.google.com/file/d/1n8oSgf...sp=sharing>

DB3000 v507 changelog: <https://drive.google.com/file/d/1myeLWr...sp=sharing>

CWDB v507 changelog: <https://drive.google.com/file/d/1nBHZT2...sp=sharing>

v1.06.XXXX

FAQs, Resources, and Learning:

CMO Architecture

Is CMO 32-bit or 64-bit? Any CMO build at 1307 or less is 32-bit. Any build above that is 64-bit. There is limited performance boost in the 64-bit environment. But 64-bit has broken the memory limitations inherent to a 32-bit app and can play much larger scenarios without crashing.

What version of Direct X is CMO built on? CMO was originally built on Direct X 9, but has recently been upgraded to Direct X 11. Any CMO build at 1307 or less is DX9. Any build above that is DX11.

Can CMO run on XOS/Mac OS? It can not run organically on any Mac OS. But it run in the Windows boot mode or Parallels on a Mac.

Is CMO able to run on Linux? Some people have gotten CMO to run on specific flavors of Linux, but it requires significant knowledge and tweaking to succeed.

What's the optimum PC configuration for CMO? It depends on what you want to do. CMO has run on very basic 8" Windows tablets with small to medium-sized scenarios and limited time acceleration. Very large scenarios with 1000s of units and 100s of missiles potentially in the air will not run at maximum acceleration even on a modern Intel i9 CPU.

What kind of GPU should I get for CMO? CMO in its basic form is not very graphics intensive. A top end GPU like an nVidia RTX 4090 will only buy you a slight increase in performance over an integrated Intel GPU chip set. But if using TacView in a large scenario, a good GPU can make the difference between a smooth experience and an unplayable scenario

Learning CMO

Is there anything I can read to prepare myself for understanding how to play CMO? Its best to start with resources already available that the devs and community have put a lot of work into for just such an occasion:

- 1) This manual: You don't have to read it. But its searchable and you should take advantage of that.
- 2) The main CMO Matrix forum: <https://www.matrixgames.com/forums/viewforum.php?f=10201>
- 3) Tutorials and videos: <https://www.matrixgames.com/forums/viewtopic.php?f=10201&t=347099>
- 4) Newer tutorials: <https://www.matrixgames.com/forums/viewtopic.php?f=10201&t=393481>
- 5) Message from the devs: <https://www.matrixgames.com/forums/viewtopic.php?f=10230&t=255656>
- 6) How to ask for help: <https://www.matrixgames.com/forums/viewtopic.php?f=10230&t=358998>
- 7) Hot Key listing: <https://www.matrixgames.com/forums/viewtopic.php?f=10230&t=280926>
- 8) New feature request ~~sub forum~~ thread:
<https://www.matrixgames.com/forums/viewtopic.php?f=10201&t=341588>
- 9) New pictures and unit descriptions:
<https://www.matrixgames.com/forums/viewtopic.php?f=10201&t=244450>

- 10) Older Mega-FAQ: This a big FAQ from the original CMNAO game. A little outdated, but you can still find useful information. https://command.matrixgames.com/?page_id=2920
- 11) The original dev website: <https://command.matrixgames.com/>

There are also a large number of player-made videos on YouTube and Rumble.

CREDITS

WarfareSims & Associates

General Director & Head of Development Dimitris Dranidis

Director of Operations & Data Administrator Paul Bridge

Development Rory Anderson, Rory Noonan, Wayne Styles, Michael McFarland

Content creation & development support Andrea Daolio, Roberto Stanza, Mark Gellis, Bart Gauvin,

Command Modern Operations

Dimitris Natsiopoulos, Zacharias Tzamargias, George Ross,
Steven Lohr, kgambit, Tomcat84, Mirabelle Benou, Al Grant,
Chris Stoner, Miguel Molina, Patrizio Ortiz, David Perez, José
Ramon García Paniagua, Clement Marti, Vyacheslav Delchev

Documentation

Colin Salt

Quality Assurance

Kobu, Maromak, Raptorx7_slith, thewood1, pjb1,
TitaniumTrout, goodwood, lowchi, zclark, mikkey, Runibl,
Kushan, kevinkins, cdcool, mfort175, Excroat3, Randomizer,
renders, xnavytc, Joelsi, erichswafford, Sigma8510, Schr75,
mahuja, AndrewJ, SakiNoE, Grackle, KC45, HellcatOne

Slitherine

Chairman

JD McNeil

Development Director

Iain McNeil

Technical Director

Philip Veale

Creative Director

Richard Evans

Marketing Director

Marco A. Minoli

Senior Producer

Bart Schouten

Producers

Tamas Kiss, Adam Grace, Erik Rutins

Product Manager
Alberto Casulini, Daniele Meneghini

Media Relations
Paolo Paglianti

Production Design
Adriana Bienati

Manual Layout
Myriam Bell

Production Lead
Matthew Ravenwood

Production Team
Joe Stephenson, Sam O'Neill
Administration

Dean Walker, Annick Tuesley
Customer Support Staff

Paulo Costa, Joseph Miller
Web Development

Valery Vidershpan, Andrea Nicola, Fernando Turi