

Acuity: RadarUI

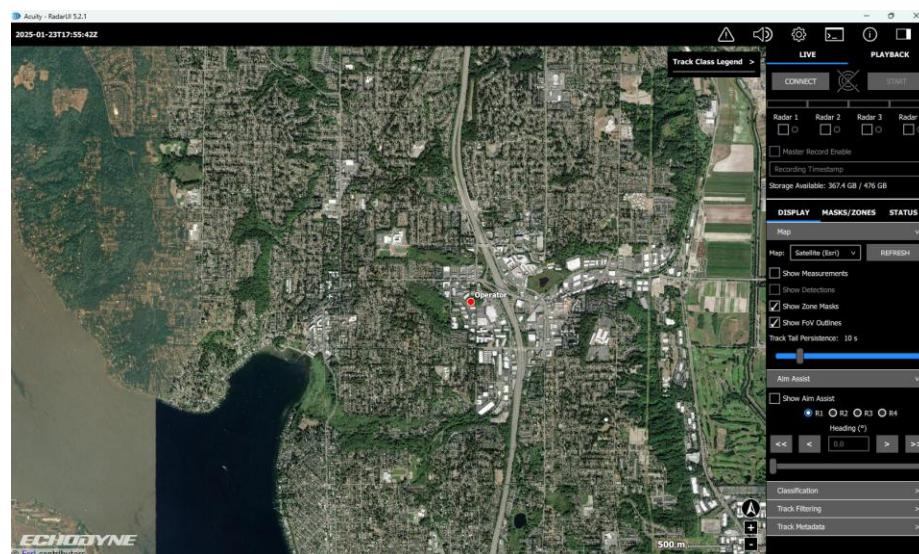
Multi-Radar User Interface

Control & GUI for All Echodyne Radars

Windows & Linux PC Application

Release Date: March 2025

Doc # 300443-006_rev01_Acuity_RadarUI-v5.2_User_Manual Released with RadarUI-v5.2



ECHODYNE PROPRIETARY - SUBJECT TO NDA OR PIA

This document contains proprietary information that shall be distributed or made available solely in accordance with a nondisclosure or proprietary information agreement. Copyright © 2025 Echodyne Corp. All rights reserved. This item is subject to U.S. export controls, and the sale, use, reexport or transfer of these items must comply with all applicable export and re-export laws and regulations, including but not limited to the Export Administration Regulations (EAR), administered by the U.S. Department of Commerce, and trade and economic sanctions, administered by the U.S. Treasury Department's Office of Foreign Assets Control (OFAC).

DISTRIBUTION NOTE



This application is provided as-is with no warranty expressed nor implied. RadarUI is provided as a demonstration, debugging and system connectivity verification tool. See official Software License Agreement PDF in 'About RadarUI' pop-up window for legal details.

TABLE OF CONTENTS

1	Overview	4
1.1	RadarUI Features	4
2	RadarUI Installation & Setup	5
2.1	Windows (Windows 10/11): Unzip Provided Folder	5
2.2	Linux (Ubuntu 20.04/22.04 LTS)	5
2.3	RadarUI - Program File Structure & Startup.....	5
2.4	RadarUI Configuration Files	6
2.5	Screen Capture Recording	6
3	Operation	7
3.1	Opening the Program and Connecting	7
4	The RadarUI Display	8
4.1	Window Controls	9
4.2	Alert Monitoring Window	9
4.3	Configuration Window.....	9
4.3.1	Configuration – ‘GLOBAL’ Settings	10
4.3.2	Configuration – RADAR TABLE	12
4.3.3	Configuration – Script Generation [EchoGuard and EchoFlight Only].....	14
4.3.4	Configuration – Extra Commands	15
4.3.5	Configuration – ‘WAYPOINT’ Settings	16
4.4	Command Line Interface Window [EchoGuard Only]	16
4.5	Informational ‘About RadarUI’ Pop-up Window.....	17
4.6	Radar Status Summary.....	17
4.6.1	Radar Status Summary – TCM State Monitoring [EchoGuard Only]	18
4.6.2	Radar Status Summary – PRISM Channels [EchoGuard Only]	18
4.7	Settings Sidebar	19
4.7.1	Sidebar Controls.....	19
4.7.2	Aim Assist.....	20
4.7.3	Classification	20
4.7.4	Track Filtering	21
4.8	Masks/Zones.....	23
4.8.1	Global Masks [EchoGuard and EchoFlight Only]	23
4.8.2	Zone Masks	23
4.8.3	Alert Zones	25
4.8.4	EchoShield Masks (EchoShield Only).....	27
4.9	Map & Data Display	28
4.9.1	Key Map Controls.....	29
4.9.2	Map Sources & Tile Caching.....	29
4.9.3	Track ‘Metadata Tag’ Toggling	30
4.10	Radar Data Playback	31
5	FAQ & Troubleshooting	33

1 Overview

RadarUI is the next generation radar visualization Human Machine Interface (HMI) tool from Echodyne. Part of Echodyne's Acuity software suite, the Windows & Linux x86 PC program provides a two-dimensional (2-D) plane view UI for Echodyne MESA (Metamaterial Electronically Scanned Array) 3-D radars. It enables parameter configuration, field-of-view (FoV) scan control, data display with geographical registration to an underlying map tile set, data recording and post-capture replay ability. It is designed primarily as an easy-to-use radar demonstration, troubleshooting and site-tuning tool to support both internal Echodyne and end-customer lab and field validation. The 64-bit program is available as a stand-alone compiled executable optimized for Windows 11 and Linux/Ubuntu 22.04 LTS on reasonably capable computers (Intel i5 or better).

RadarUI communicates with the radar through the integrated BNET and Librio API libraries, interface programs that handle the Ethernet TCP/IP communications and bundles the streaming data into buffers that are provided to RadarUI upon request. The library APIs also provide the option to record radar data directly to disk while simultaneously buffering that data to RadarUI or other programs for visualization.

1.1 RadarUI Features

RadarUI was designed to support multiple radars (up to 4) and facilitate system control and interaction with an accessible and intuitive graphical user interface. Radar parameters that are commonly adjusted by the user are exposed in graphical buttons and text boxes – the user does not need to know the low-level radar commands nor their specific syntax. More advanced configuration commands used to tune the radar for specific scenarios are accessible using an 'Extra Commands' window automatically configured from the application during the start of radar operation.

RadarUI has a wide set of features that support optimization for multiple scenarios. These features, explained later in this manual, are divided into the following key categories:

- Radar parameter configuration and set-up including Mission Set presets
- Radar operation and status monitoring
- Radar data display of live tracks and related metadata fields in real-time
- Radar data display of lower-level detection and measurement data ports
- Registration of radar data to worldview maps (street or satellite) from ESRI™ or Mapbox™
- Radar data recording and logging of all data port types to disk
- Recorded data playback with speed, jump and loop controls
- Generation of radar configuration scripts for 3rd party integrations
- Video screen capture generation (*via Windows Game DVR feature, OBS or similar 3rd party app*)

2 RadarUI Installation & Setup

2.1 Windows (Windows 10/11): Unzip Provided Folder

RadarUI is compatible with Windows 10 and Windows 11. The current version of RadarUI does not require a dedicated program installer and is easy to deploy with minimal set-up. The provided ~32 MB .zip file decompresses to a ~73 MB folder which may be placed wherever desired without requiring administrative rights or modifying any registry settings.

Inside the program folder are many files supporting the application. The only ones to be concerned with are the **radarui.bat** application batch file and configuration files as explained below. A shortcut to the **radarui.bat** file may be located on the desktop for easy access.

2.2 Linux (Ubuntu 20.04/22.04 LTS)

RadarUI is compatible with Ubuntu 20.04 LTS and Ubuntu 22.04 LTS. Other Ubuntu versions (18.04) are not supported. Other Linux flavors (e.g. Fedora, Arch) are not supported.

On a clean Ubuntu 22.04, it may be necessary to install Perl libraries (libpcre2-16.so.0). This can be done by executing the following commands:

```
sudo apt -y install libpcre2-16-0.
```

Second, to install the program, simply extract the provided archive in the location of your choice and run the radarui.sh script to start the program.

2.3 RadarUI - Program File Structure & Startup

An example of how the application is organized is shown below. Start RadarUI by opening the radarui.bat file:

/200021-009_rev02_radarUI-v5.x	
└/.cred-store	- Contains super-user credentials (this is a hidden folder)
└/bin	
└/bin	- Binary folder structure with executable & many DLLs
└/log	- Text log files useful for debugging unexpected behavior
└/log_files	- Contains log files
└/tiles	- Folder containing all map tiles (100 MB maximum)
└/Toughbook-lib	- Contains items specific to Panasonic Toughbook laptops
└/Documents	- Folder containing User Manual and License documents
└/RadarConfig	- Folder containing all radar configuration files
└RadarConfig_location_1.json	- Example radar config for sample location #1
└RadarConfig_location_2.json	- Example radar config for sample location #2
└/RadarData	- Location where radar data will be saved (configurable)
└/MESA-900123	- Example EchoGuard radar data folder
└/EchoShield-0000	- Example EchoShield radar data folder
└/radarui_data	- Folder containing timestamped RadarConfigs for each collect
└/radarui.bat	- Batch file (Windows ver) to <u>start program</u> (points to executable in /bin)
└SysConfig.json	- Program system configuration file

2.4 RadarUI Configuration Files

RadarUI key application settings and radar configuration parameters are stored in **SysConfig.json**, **RadarConfig.json** files, which are located in the file tree shown above. For typical usage, there is no need to directly edit the JSON formatted files as the program manages the settings through these files. For the advanced user, the RadarConfig.json file is human readable and can be edited if desired. Care should be taken to back-up these files as they contain everything necessary to define the radar operation and RadarUI rendering of the radar data on the map.

Note: ***malformed file formats will be noted and overwritten*** with a default configuration file when detected and the corrupt file will be saved with a timestamp for later review and correction. JSON is an easy to view and edit file format, but is very sensitive to bracket and comma usage. It is recommended to use a programming editor (e.g. [Visual Studio Code](#)) with JSON file linting to detect and highlight formatting errors if editing directly.

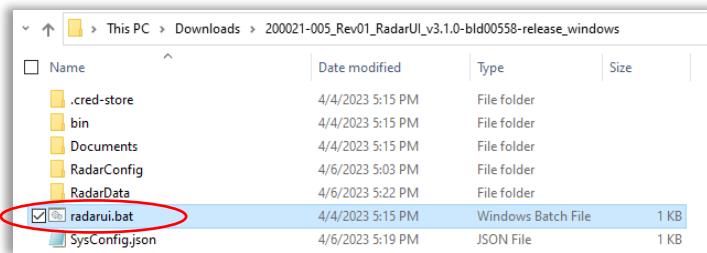
Each time a radar collect is recorded, a copy of each .JSON configuration files is made with an appended timestamp for associating with the recorded data to give it context for playback.

2.5 Screen Capture Recording

RadarUI does not natively support screen recording, but it's directly compatible with the Windows 10 'Game Bar' functionality that enables high quality screen recording. To activate, press **Windows + G** keys together, identify the UI application as the desired 'game' and select the record function. Screen recordings will be saved to your local /Video/Captures folder. Other 3rd party solutions such as OBS on Windows & Linux also exist to support recording if desired.

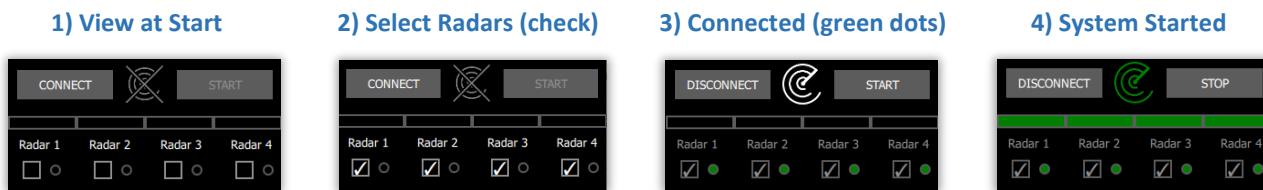
3 Operation

To start the program, double-click on the **radarui.bat** file to launch the program. No installer is required. This allows you to have multiple versions of the program available if desired.



3.1 Opening the Program and Connecting

The following series of images shows the RadarUI state progression for starting a multi-radar installation using the GUI controls.



At program start (1), all radars are unselected, and nothing is connected, nor transmitting. The user selects/deselects which radars are to be connected using the checkboxes (2), taps the 'CONNECT' button and can observe the radars transitioning to the TCP/IP connected state as indicated by the green/yellow/red indicators to the right of the checkboxes. In the connected state (3), the 'System Transmitting Indicator' transitions from greyed-out to white, indicating that it is now possible to start the radars. If a particular link is not stable, the user will observe the indicator changing between red and green as RadarUI attempts to recover the connection.

When ready to operate the radars, the user selects the 'START' button and the currently connected radars will be commanded to start operating in Search-While-Track (SWT) mode (4). The 'System Transmitting Indicator' and respective 'Individual Radar Transmitting Indicators' will transition to the green active state indicating RF energy is being radiated from the antennas. Upon reaching State (4) the user should observe tracks forming on the map display and updating FPS values in the Radar Status Summary bar.

Radar Operation Notes:

1. During radar operation, be sure to observe the transmit aperture safe distance for all personnel.
2. If at any time during operation RadarUI loses TCP/IP connection with the transmitting radars, it will attempt to reconnect the command port. This can be observed in the red/green 'Connection Indicator' next to the check boxes. However, for safety reasons the radar will not automatically resume transmitting. This can be observed by the 'Individual Radar Transmitting Indicator' illuminated as red (transmit blocked) instead of green (transmitting). To re-enable, toggle the STOP/START button.

4 The RadarUI Display

The display layout assumes a minimum monitor resolution of 1280 x 1024 pixels and is optimized for HD resolution displays (1920 x 1080 pixels) for best rendering of the detailed real-time radar data, with up to 150% zoom supported. RadarUI also supports 4K monitors at 150% zoom.

The primary screen is divided into four key interaction areas:

- Window Controls (top right)
- Settings Sidebar (right)
- Radar Status Bar (top left)
- Map & Data Display (center)

The **Window Controls** region enables the presentation of additional screens including an Information/About pop-up, a Command Line Interface (CLI) window, alert monitoring window and the Settings Sidebar.

The **Settings Sidebar** is the primary interaction area for connecting, starting, stopping and monitoring the radar transmit operation. The lower portion of the sidebar contains UI controls for modifying the map, enabling overlays, track filtering, customizing track metadata, adding custom masking zones, etc.

The **Radar Status Summary** bar at the top shows current UTC (Z=Zulu) time as well as key status information for each operating radar to confirm correct operation modes, reasonable frame rates and radar orientation.

The **Map & Data Display** depicts the radar data aligned to a high-fidelity world map with standard zoom and rotation controls. Track data is rendered in real-time with associated metadata available for inspection. Lower-level radar detection and measurement data can be enabled to support troubleshooting difficult environments.

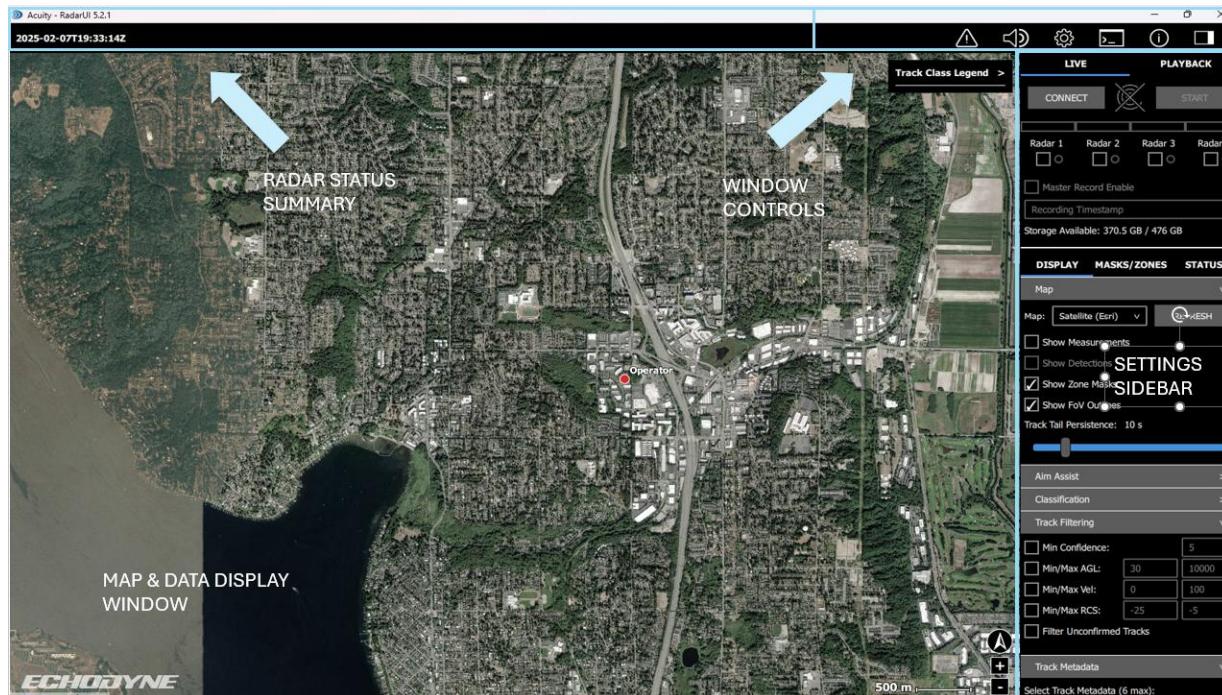


Figure 1 – RadarUI Key Interaction Regions

4.1 Window Controls

From left to right, the Window Icon Controls are: Alert Monitoring Window, Alert Monitoring Audio Mute/Unmute, Configuration Settings, Command Line Interface, Information About, and Settings Sidebar Show/Hide. Touching or mouse clicking on each icon will trigger the associated window.



4.2 Alert Monitoring Window



The Alert Monitoring Window is accessed by pressing the alert icon in the Windows Controls bar. For EchoShield radars, RadarUI monitors the SBIT (startup built-in test) and CBIT (continuous built-in test) status available in the Status packet and if any radar alerts are detected, RadarUI provides both a visual and audible alert for the detected fault(s). The window will display the fault(s) detected.

The speaker icon next to the alert icon provides the option to mute or unmute audible alerts. In the Alert Window the audible mute/unmute option is also available along with the option to mute/unmute the alert window.

Note that the alert icon in the Windows Controls bar changes appearance when there is a warning or fault. Thus, even if both audio and visual alerts are muted, one can still see if there is an alert by the state of the alert icon.

4.3 Configuration Window



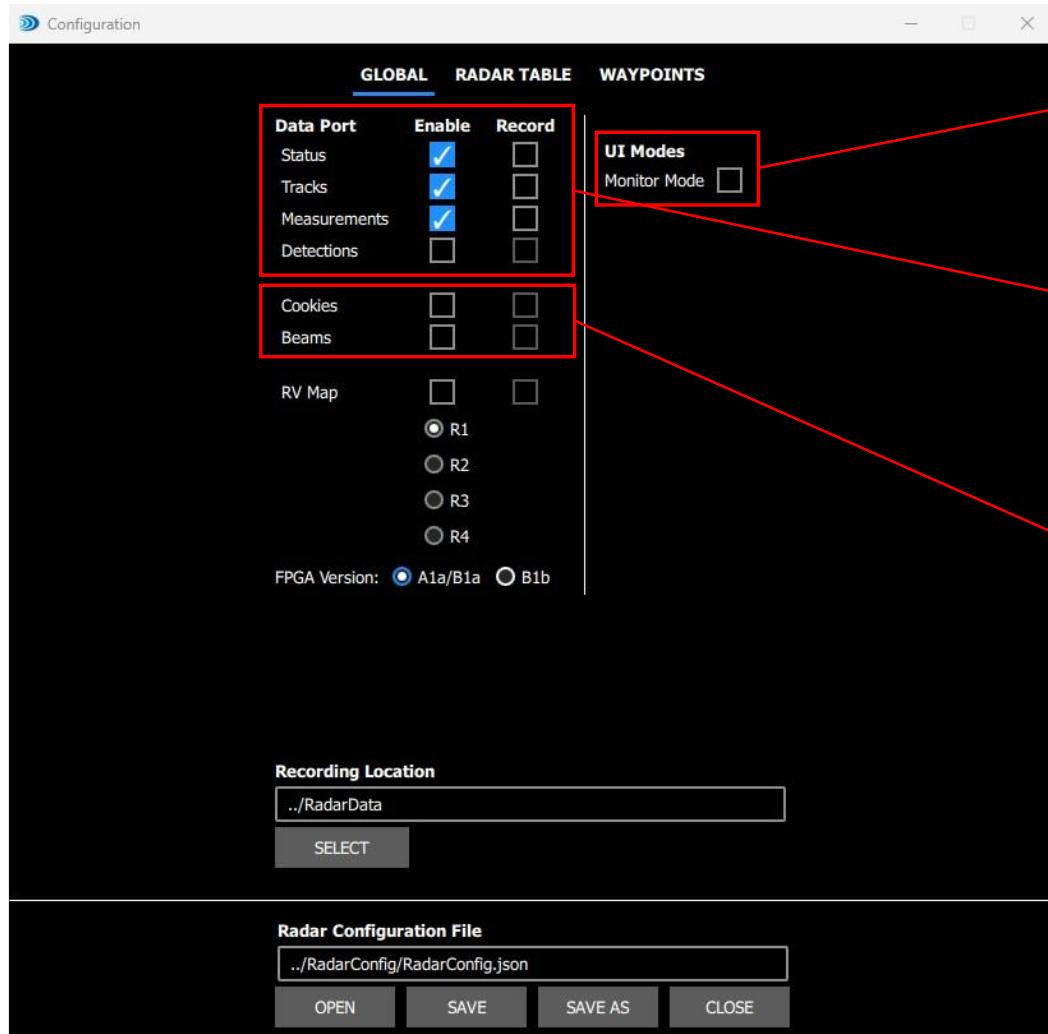
The Configuration window is accessed by pressing the gear icon in the Window Controls bar. This window is the key area for configuring the radar data recording, static Ipv4 address, physical location/orientation, radar scan FoV and optional map waypoints. One can scroll through the various sub-windows by selecting the desired tab (e.g., 'Global', 'Radar Table') and observing the blue underline move in response to the selected sub-window.

The bottom of the Configuration window allows one to Open or Save a custom Radar Configuration File (RadarConfig.json file) that captures all of the presets in this Configuration window for later use.

Note – The Configuration window may be opened and reviewed at any time, but radar configuration fields may only be actively edited when not operating the radars (via Settings Sidebar button). A red text warning '**Some fields are READ ONLY. DISCONNECT for more functionality**' will be presented above the Radar Configuration File section in this 'view only' state. Radar Orientation for EchoGuard radar can be edited while operating.

4.3.1 Configuration – ‘GLOBAL’ Settings

This sub-window allows for selection of which radar data ports you want to connect or ‘enable’ and for those connected ports, which radar data to record to disk (optional).



UI Modes: Monitor Mode

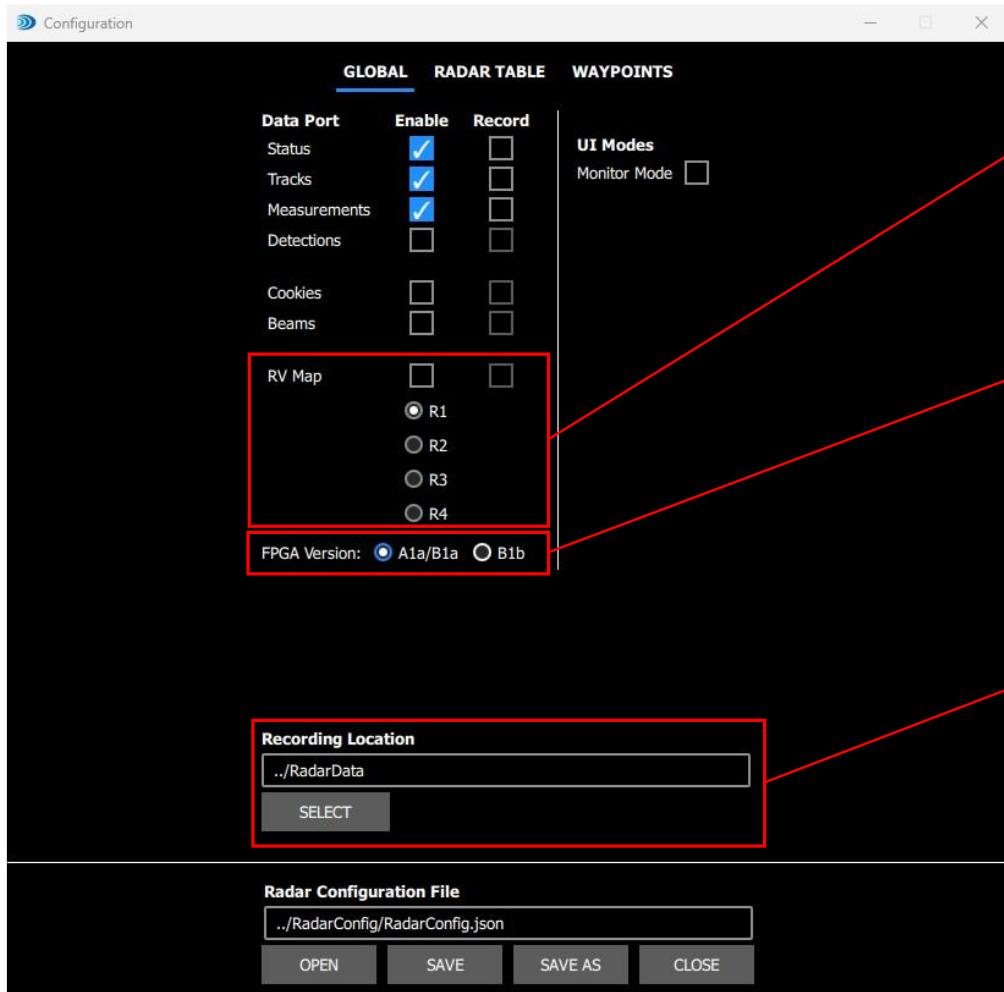
- Allows users to display data only from radars in use by others.

Data Port Selection for Display and Recording

- These controls allow the user to choose which radar data ports have data displayed in RadarUI.
- The 'Measurements' and 'Detections' let users see the raw data and processed detections that feed the tracker associations.
- If more than one radar is in use, the same enable and recording settings will be applied to all connected radars

Cookies, Beams

- These controls only apply to EchoShield radars.
- The 'Cookies' and 'Beams' controls allow users to record that data.



RV Map

On EchoGuard and EchoFlight, recording of RVmaps may also be enabled on this screen, but they consume a **large** amount of TCP/IP bandwidth and disk space and are disabled by default. [Only enable this port after consultation with the radar manual and review of available disk space and sufficient Ethernet connection data rate.](#) Only one radar RVmap stream at a time may be recorded due to Gbit bandwidth limitations. The track port must also be enabled to record RVmaps.

FPGA Version

- Applies to EchoGuard radars only.
- Used to set FOV min/max range limits.

Recording Location

- The recording location may be customized to any location on your local machine, but defaults to the program install directory's "./RadarData" folder.
- For EchoGuard, recorded data will be organized by radar serial number and timestamped subfolder each time you 'Start' the radars and begin transmitting.
- For EchoShield, recorded data is organized first by timestamped folders and then by radar serial number.
- Saving to a networked drive may have reduced performance and potentially dropped data packets depending on the number of radars, data ports selected, and network throughput.
- To end recording, simply push the 'Stop' button and the radar(s) will cease transmitting and radar data recording will complete. The radar data is recorded in a 'chunked' manner where individual packets are concatenated into larger binary files to minimize file system I/O overhead so upon STOP, the last file 'chunk' will be written.

4.3.2 Configuration – RADAR TABLE

This sub-window allows for customization of each connected radar (up to 4) in an easy to compare table. If you only have one radar in use, then begin with Radar 1, if configuring two radars, then use this table to set-up Radars 1 & 2, etc. **NOTE – This table is scrollable so be sure to pan down for additional radar configuration fields.**

The screenshot shows a software window titled "Configuration". At the top, there are three tabs: "GLOBAL", "RADAR TABLE" (which is selected), and "WAYPOINTS". Below the tabs is a table with four columns labeled "Radar 1", "Radar 2", "Radar 3", and "Radar 4". The rows represent different configuration parameters. The table data is as follows:

	Radar 1	Radar 2	Radar 3	Radar 4
Label	R1	R2	R3	R4
IPv4 Address	10.0.83.40	169.254.1.11	169.254.1.12	169.254.1.13
Product Mode	EchoShield	v	EchoGuard	v
Mission Set	C-UAS_1	v	Pedestrian	v
Sync GPS	SYNC	SYNC	SYNC	SYNC
Latitude (°)	47.708900	47.708870	47.708870	47.708870
Longitude (°)	-122.187393	-122.187393	-122.187393	-122.187393
Altitude (m)	63.0	10.0	10.0	10.0
Heading (°)	186.8	10.0	20.0	30.0
Sync IMU	SYNC	SYNC	SYNC	SYNC
Pitch (°)	20.0	0.0	0.0	0.0
Roll (°)	0.0	0.0	0.0	0.0
Range Min (m)	100	21	21	21
Range Max (m)	2000	2000	2000	2000
Search Az Min (°)	-60	-60	-60	-60
Search Az Max (°)	60	60	60	60
Search El Min (°)	-24	-24	-24	-24
Search El Max (°)	40	12	12	12

At the bottom of the window, there is a section for "Radar Configuration File" with a text input field containing "./RadarConfig/RadarConfig.json" and four buttons: OPEN, SAVE, SAVE AS, and CLOSE.

Figure 2 – Configuration Table: Radar Settings

Customizations include:

- Radar network location (IPv4 Address)
- Radar Mission Set – EchoGuard: cUAS, Pedestrian, Plane/GB-DAA. EchoShield: C-UAS_1, C-UAS_2, Coastal C-UAS, Dismount. See respective radar manual for definitions.
- Product Mode (EchoShield, EchoGuard, EchoGuard-CR)
- Radar physical X/Y/Z position (Latitude, Longitude, Altitude)
 - Lat/Long in decimal degrees, Altitude in meters
 - **On EchoShield, Altitude should be input in orthometric height / MSL (mean sea level) for features like AGL filtering to work properly.** On EchoGuard and EchoFlight, altitude of targets is reported relative to the radar. It's assumed the radar is at the Altitude value specified, so it's user preference whether this is MSL or something else.
- Radar physical angular orientation (Heading/Yaw, Pitch, Roll) Heading in true North for correct map alignment.
- (EchoGuard only) The **SYNC IMU** button initiates a 5 second averaging of the internal radar roll & pitch to auto-populate those fields.

- Radar Search & Track Field-of-views (FOVs)
- Radar minimum & maximum range (sets min/max range mask parameters)
 - On EchoShield, minimum range is not configurable from default setting.
- Frequency channel per radar
- (EchoGuard only) TCM and PRISM channelization per radar
- Additional radar parameter settings may be called via the 'Extra Commands' dialogue box. Enable w/check box. Select 'Edit' to customized commands.

Notes on radar configuration:

- 1) If the radars are operating (pressed START), you won't be allowed to edit certain radar specific fields until you 'STOP'. This is indicated by red text indicating viewing mode only.
- 2) Each of the fields offers helpful tooltips and input range checking. You will be prompted if a value is out of range or you forget to save before closing the window.
- 3) [EchoGuard Only] When Connecting to the radar(s), the RadarUI will configure the radar(s) in the following order:
 - a. Issue a RESET:PARAMETERS command to establish known configuration.
 - b. Set Mission Set defaults
 - c. Set Search & Track FOVs
 - d. Set Min & Max Range Masks
 - e. Set channelization (Frequency band, TCM & PRISM)
 - f. Set Global and Zone Masks
 - g. Set Extra Commands' provided in dialogue box.
- 4) To prevent conflicting with the RadarUI normal operation, avoid using any extra commands that the UI sends (EchoGuard e.g., OPERATION:MODE, SWT:SEARCH & SWT:TRACK FOVs, RANGE:MASK, etc.).
- 5) Annex H of the ISO-6709 standard is used to set and read the geographical positions. ISO-6709 is the international standard for representation of latitude, longitude, and altitude for geographic locations:
 - a. **Sign conventions:** North latitude and East longitude are positive (+). South latitude and West longitude are negative (-).
 - b. RadarUI only supports decimal degrees for latitude & longitude. The position is only given in whole and decimal degrees. The integer portion is fixed length, using 2 digits (-90 to +90 degrees) for latitude and 3 digits (-180 to +180 degrees) for longitude. The fractional part must have the appropriate number of digits to represent the required precision of the coordinate. Five (5) decimal places is roughly 1-meter precision and sufficient for radar usage.

4.3.3 Configuration – Script Generation [EchoGuard and EchoFlight Only]

Once the radar configuration is satisfactory, RadarUI can generate a complete radar set-up script that captures all the configuration commands the UI is sending to the radar. This is useful for 3rd party integrators to leverage the Echodyne UI for initial radar site-tuning, then auto-generate the final configuration script.

To do this, open the configuration window, select the Radar Table tab, then scroll down and click ‘Generate’. The script is broken into three sections: 1) A commented-out header ‘##’ (ignored by BNET CLI/DLL) that captures the physical orientation of the radar, 2) UI set-up commands that configure the operation mode, FOV, channelization and defined masks and 3) User-specific ‘Extra Commands’ entered via the dialogue box.

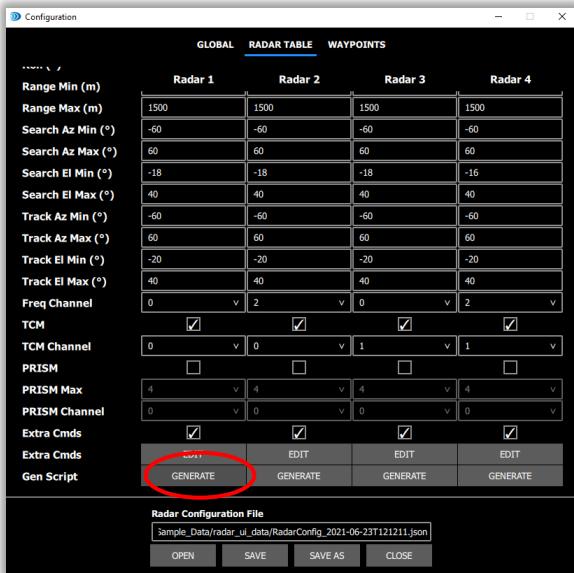


Figure 3 – Radar Table Configuration Menu

```

1 ## This script is auto generated from RadarUI for external radar setup
2 ## Radar Config Version: 1.7.0
3 ## Radar Label: R1
4 ## IPv4 Address: 169.254.1.10
5 ## Latitude: 47.7089
6 ## Longitude: -122.187
7 ## Altitude(meters): 10
8 ## Heading(degrees): 0
9 ## Pitch(degrees): 0
10 ## Roll(degrees): 0
11 RESET:PARAMETERS
12 *IDN?
13 SYSPARAM? eldorado
14 MODE:SWT:OPERATIONMODE 0
15 MODE:SWT:SEARCH:AZFOVMIN -60
16 MODE:SWT:SEARCH:AZFOVMAX 60
17 MODE:SWT:TRACK:AZFOVMIN -60
18 MODE:SWT:TRACK:AZFOVMAX 60
19 MODE:SWT:SEARCH:ELFOVMIN 0
20 MODE:SWT:SEARCH:ELFOVMAX 40
21 MODE:SWT:TRACK:ELFOVMIN -40
22 MODE:SWT:TRACK:ELFOVMAX 40
23 RANGE:MASK eldorado 0, 128, 134, 0, 31
24 RANGE:MASK eldorado 1, 742, 2047, 0, 31
25 DMS:CHANNEL 0
26 TIME:CHANNEL:ENABLE eldorado FALSE
27 PRISM:CHANNEL:ENABLE eldorado FALSE
28 ZONE:MASK:CLEAR eldorado
29 ZONE:MASK:ENABLE eldorado FALSE
30 PLATFORM:STATE:ORIENTATION eldorado 0.000000, 0.000000, 0.000000
31 PLATFORM:STATE:ELEVATIONAGL eldorado 10.000000
32 RSP:CLUTTERMASK:WIDTH eldorado 3
33 RSP:RCMASK:MINRCS -30
34 RSP:RCMASK:MAXRCS 100
35 AGLMASK:MINAGL eldorado -10000
36 AGLMASK:MAXAGL eldorado 10000
37 AGLMASK:ENABLE eldorado FALSE
38 ## Commands below are additional commands set by the user
39

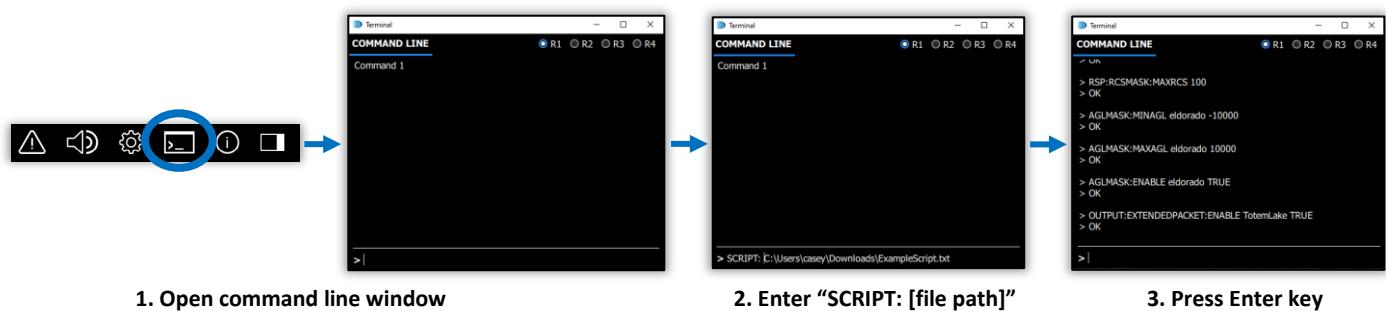
```

Figure 4 – Example Generated Script

Note: Generating a script with radars using software version 17.0 or newer, then executing that script on a radar with an older software version, will throw an error. This is because older radar software expects a password for the extended track packet that is no longer required in version 17.0 and onward.

4.3.3.1 Running a Script

To run a script, open the command line window by clicking the icon at the top right, and use the command “SCRIPT:” followed by the location path of the generated script file. Press Enter to run the script, which will configure the radar as it was when the script was generated.



4.3.4 Configuration – Extra Commands

The Extra Commands window allows for additional radar parameter settings to be customized.

4.3.4.1 EchoGuard and EchoFlight

When interfacing with an EchoGuard or EchoFlight radar, the syntax is the same as that used in the BNET CLI application and the integrated RadarUI CLI.

4.3.4.2 EchoShield

When interfacing with an EchoShield radar, the syntax is <parameter_group>:<parameter>:<value>. The ‘parameter_group’ is the group of commands identified in the EchoShield Developer Manual for parameters, which can be either “radar,” “network,” or “system”. Likewise, ‘parameter’ is the name of the command, and ‘value’ is the value input for that command. Note, nested parameter types are not supported, nor are array parameters; only simple name-value pairs.

For example, to set the minimum azimuth search field of view to -40 degrees using the “search_azmin” command in the Extra Commands window, the syntax would be “radar:search_azmin:-40”. See Figure 6 below.

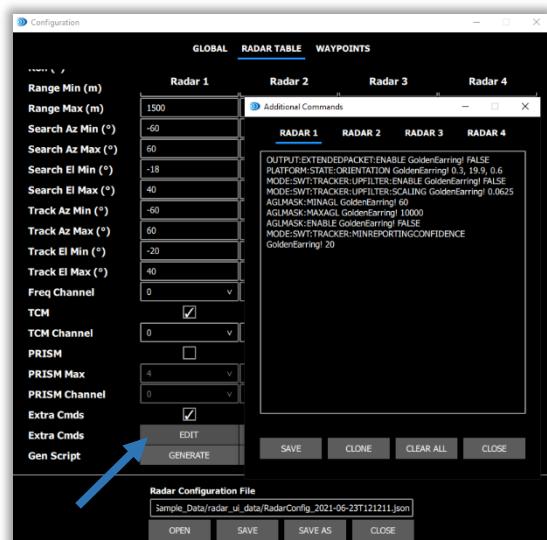


Figure 5 – EchoGuard Example Extra Commands

Parameter Group (“radar”, “network”, or “system”)	
Parameters	
Parameter Attributes	
System Parameters	
Network Parameters	
RadarApp Parameters	
rotation_relative_to_reference_yaw	
rotation_relative_to_reference_pitch	
rotation_relative_to_reference_roll	
position_relative_to_reference_x	
position_relative_to_reference_y	
position_relative_to_reference_z	
search_azmin	
search_azmax	
search_elmin	
search_elmax	
track_azmin	

Parameter	
Attribute	Value
Persistence	Volatile
User Access	USER (default)

Minimum azimuth search beam angle in degrees. Special constraint: search_azmin < search_azmax.

Parameter	
Attribute	Value
Default	-65
Type	number
Constraints	-65 ≤ x ≤ 65
Comments	Modifiable Mode Operation and Service
Persistence	Volatile
User Access	USER (default)

Value

Table 109 – continued from previous page

Parameter	
Attribute	Value
Persistence	Volatile
User Access	USER (default)

search_azmax

Maximum azimuth search beam angle in degrees. Special constraint: search_azmin < search_azmax.

Parameter	
Attribute	Value
Default	65
Type	number
Constraints	-65 ≤ x ≤ 65
Nullable	No
Modifiable Mode	Operation and Service
Persistence	Volatile
User Access	USER (default)

search_elmin

Minimum elevation search beam angle in degrees. Special constraint: search_elmin < search_elmax.

Parameter	
Attribute	Value
Default	-40

Figure 6 – Command Parameter Group, Parameter, and Value in the EchoShield Developer Manual

4.3.5 Configuration – ‘WAYPOINT’ Settings

This sub-window allows for the definition of up to four (4) custom waypoints or Points of Interest (POIs) on the map depicted as red circles. Each waypoint may have a custom 8-character label as well as latitude and longitude location specified in decimal degrees. If you do not define a waypoint, it will not be shown on the map. These waypoints provide a means to show locations of other assets, primary and secondary radar locations, etc.

The **first waypoint (#1) is special** in that it defines the program start location for the center of the map. It is recommended that you locate this waypoint somewhere in the region of your radar constellation so the program will auto-locate to your radar FOVs. You may choose to locate this at the primary radar location, a separate C2 location or other designated location.

Removing the waypoint label text disables the corresponding waypoint from the map view. Pressing ‘COPY’ will bring over the corresponding (latitude, longitude) pair from the respective radar (e.g., Radar 1 -> Waypoint 1).

4.4 Command Line Interface Window [EchoGuard Only]

This movable and resizable floating command window allows for a command line interface (CLI) interaction with the connected radars similar to the BNET executable. Select the radio button for the radar of interest and type commands at the bottom prompt ‘>’. Entered commands and radar responses will be shown on the screen. This window is helpful when trying to determine if configuration scripts have loaded correctly. Simply enter SYSPARAM? <su_pswd> and a system parameter state will be dumped to the screen. The up/down arrows allow you to access your typed command history and the window supports scrolling. If multiple radars are connected, use the radio buttons to communicate with the desired radar.

Note – When the radar(s) are transmitting, only a subset of commands is allowed. Available commands in the current operation mode can be displayed by entering the **LIST** command.

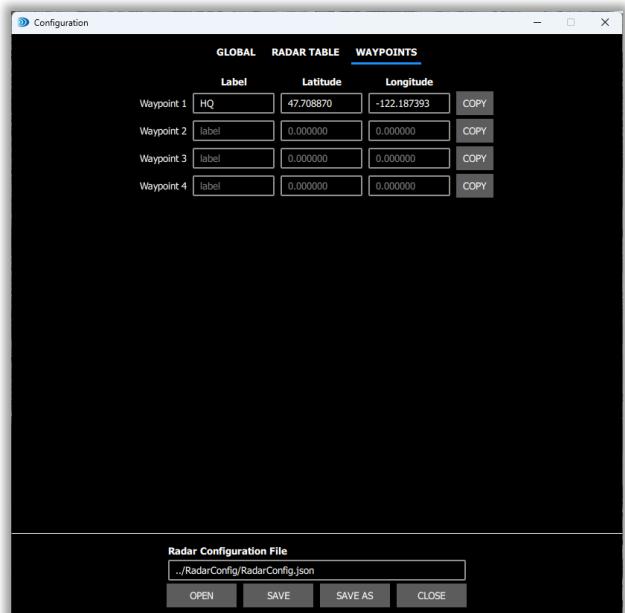


Figure 7 - Configuration: Waypoint POIs

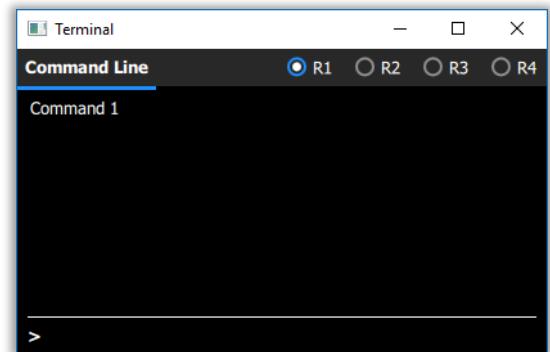


Figure 8 - Command Line Interface

4.5 Informational ‘About RadarUI’ Pop-up Window

This icon reveals the program information window which contains:

- Program version information & copyright (Windows10 version)
- Company website link: www.echodyne.com
- Support email link: support@echodyne.com
- Local PDF file hyperlinks to the:
 - **User Manual PDF (this document)**
 - **Software Distribution License PDF**

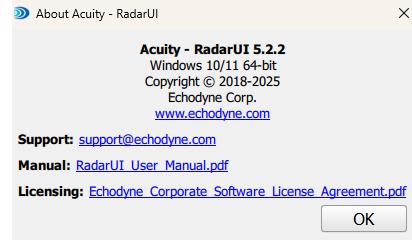


Figure 9 - About Dialogue Box

4.6 Radar Status Summary

The display’s top left contains an informational status bar that shows system time (UTC) and details related to actively scanning radars.

For each active radar (e.g., R1, R2, ...), the status bar displays configured operation modes (e.g., cUAS), number of active of tracks, and real-time Search FOV frame rates (e.g., 1.9 fps). Hovering momentarily with the mouse over the radar label (e.g. ‘R1: cUAS’) will activate a tooltip that shows additional details for that particular radar. Information in the tooltip includes system operating state (e.g. SWT = Search-While-Track mode), negotiated Ethernet link speed (e.g. 1 Gbit/s), Channelization status (Frequency, TCM, PRISM) and unit orientation. This information is extracted from the status packets and updated at the status packet update rate which defaults to 20 Hz, but can be adjusted lower using a radar command (see respective radar manual).



Figure 10 - Radar Status Pop-up Window

The displayed date and timestamp are based on the host PC clock, which is assumed to be accurate via NTP service or GPS synchronization. RadarUI always converts the local PC time to Coordinated Universal Time (UTC) for a globally unambiguous reference, and displays in an ISO-8061 compliant format. For EchoGuard and EchoFlight, this time reference is used for setting the internal radar system clock(s) upon connecting with each radar, and all radar data packets will have this timestamp embedded in their header fields. See the EchoGuard/EchoFlight User Manual for additional detail on these packet formats and initializing the radar system clock. EchoShield radars have their own time synchronization to NTP or GPS, thus RadarUI does not set the internal EchoShield radar system clock.

All RadarUI-created data folders and RadarConfig_<timestamp>.json snapshots for playback will have a date and timestamp set to the current PC time zone value. If one desires these file and folder timestamps to correspond to the underlying UTC data packet timestamps, simply set your PC time zone to UTC/GMT.

Label	Radar 1	Radar 2	Radar 3	Radar 4
IPv4 Address	169.254.1.10	169.254.1.11	169.254.1.12	169.254.1.13
Product Mode	EchoGuard	EchoGuard	EchoGuard	EchoGuard
Mission Set	Pedestrian	Pedestrian	Pedestrian	Pedestrian
Sync GPS	SYNC	SYNC	SYNC	SYNC
Latitude (*)	-47.708870	-47.708870	-47.708870	-47.708870
Longitude (*)	-122.187393	-122.187393	-122.187393	-122.187393
Altitude (m)	10.0	10.0	10.0	10.0
Sync IMU	SYNC	SYNC	SYNC	SYNC
Heading (*)	0.0	10.0	20.0	30.0
Pitch (*)	0.0	0.0	0.0	0.0
Roll (*)	0.0	0.0	0.0	0.0
Range Min (m)	21	21	21	21
Range Max (m)	2000	2000	2000	2000
Search Az Min (*)	-60	-60	-60	-60
Search Az Max (*)	60	60	60	60
Search El Min (*)	-24	-24	-24	-24
Search El Max (*)	12	12	12	12

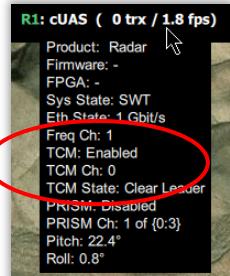
Radar Configuration File
RadarConfig/RadarConfig.json
OPEN SAVE SAVE AS CLOSE

The radar’s pitch and roll angles are provided by the internal IMU which senses the gravity acceleration vector and provides degree accurate values that may be used to manually update the Radar Configuration fields. Use the SYNC IMU button in the Configuration window dialogue to automatically average (5 seconds) these sensors to set a nominal roll & pitch of the radar. Currently this feature is only supported for EchoGuard radars.

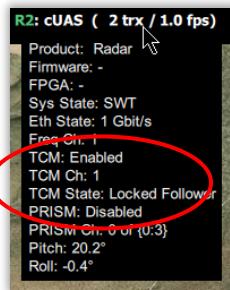
4.6.1 Radar Status Summary – TCM State Monitoring [EchoGuard Only]

When using Time Channels via the TCM functionality, the UI also provides state monitoring and color-coded indication that the system channelization is operating nominally.

For TCM Channel 0, the radar 'R1' should be the leader when operating in SWT Mode and indicate as such via the TCM State field as 'Clear Leader'. In this case, **R1** is color-coded green indicating that the system is operating nominally. If R1 was not able to establish itself as a clear leader or entered an error state, the radar label would be color-coded red as **R1**.

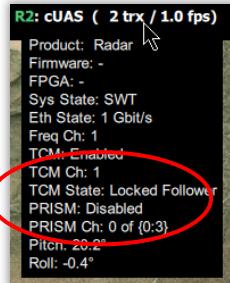


Similarly, for the TCM Channel 1 radar (R2 in this example) the correct state for this radar when operating in SWT Mode is 'Locked Follower'. Inspection of the drop-down menu indicates that this is indeed the case and **R2** is also color-coded green.



4.6.2 Radar Status Summary – PRISM Channels [EchoGuard Only]

PRISM channelization is a 3rd radar deconfliction option complementary to Frequency and TCM channels. RadarUI now supports easy enabling and configuration of PRISM channels via the Configuration window as well as monitoring in the Radar Status Summary dropdown. Note, radars must observe the correct spacing as covered in the Radar User Manual and training. Also, radars with a -00XXXX serial number (older than March 2020) must have the A6 FPGA upgrade to use PRISM. Radars with a -000XXXX can use PRISM out of the box.



4.7 Settings Sidebar

The Settings Sidebar contains key User Interface (UI) controls for selecting, connecting, and starting the radar units as well as customizing the display of various map elements.

The sidebar is broken into control sections for:

- LIVE operation vs recorded PLAYBACK operation
- Selecting, Connecting & Operating (i.e., Enable Transmitting) Radars (up to 4)
- DISPLAY Sub-window Options:
 - Selecting the base map, map refreshing, showing optional features (e.g., aim assist)
 - Configure how tracks are classified and displayedAdjusting the persistence of the track tail from 5 to 60 seconds
 - Enabling and setting track filtering options
 - Selecting the desired default metadata fields (up to 6 max)
- MASKS/ZONES Sub-window Options:
 - Set global masks. Create, edit and manage zone masks over the field of view.

4.7.1 Sidebar Controls

Shown below are descriptions of the user interface controls and their groupings.

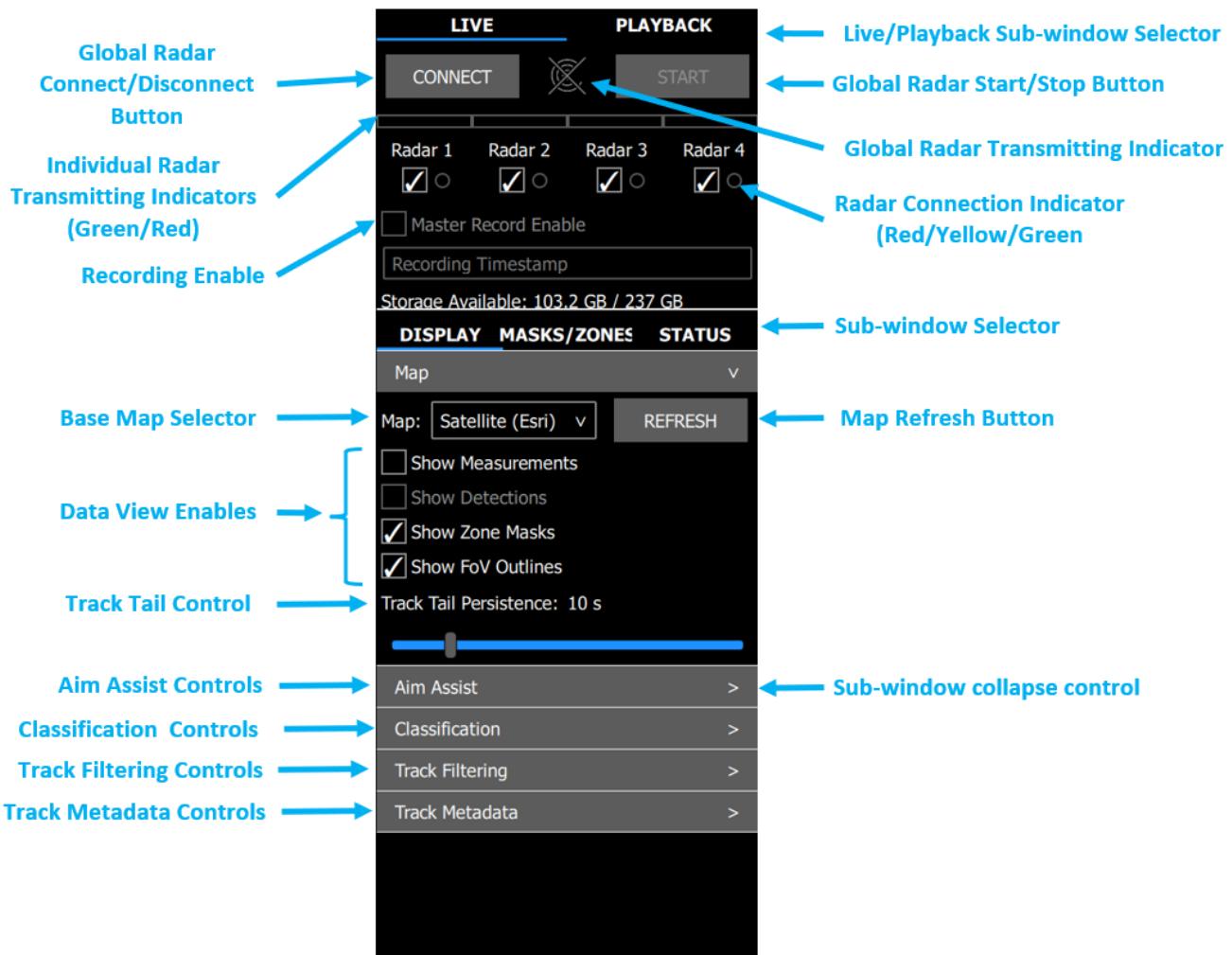


Figure 11 - Settings Sidebar: DISPLAY Controls

4.7.2 Aim Assist

Aim assist helps fine-tune the yaw/heading of each radar for optimal geo-alignment to the underlying map. When the checkbox is enabled, a red line is shown within the radar FOV that represents the broadside (0° Azimuth beam) and enables controls to adjust the heading. The slider and arrow controls allow fine tuning of each radar individually down to 0.1 degree. Since map display is just transforming the raw radar X/Y/Z track information to world coordinates, aim assist is editable while the radar is operating. Once alignment is satisfactory, you can deselect the checkbox and close the aim assist sidebar. The RadarConfig.json file will be updated with this new heading for future operation. For precision alignment, Echodyne offers a radar mounting bracket accessory with integrated red dot site that can be used in conjunction with aim assist.

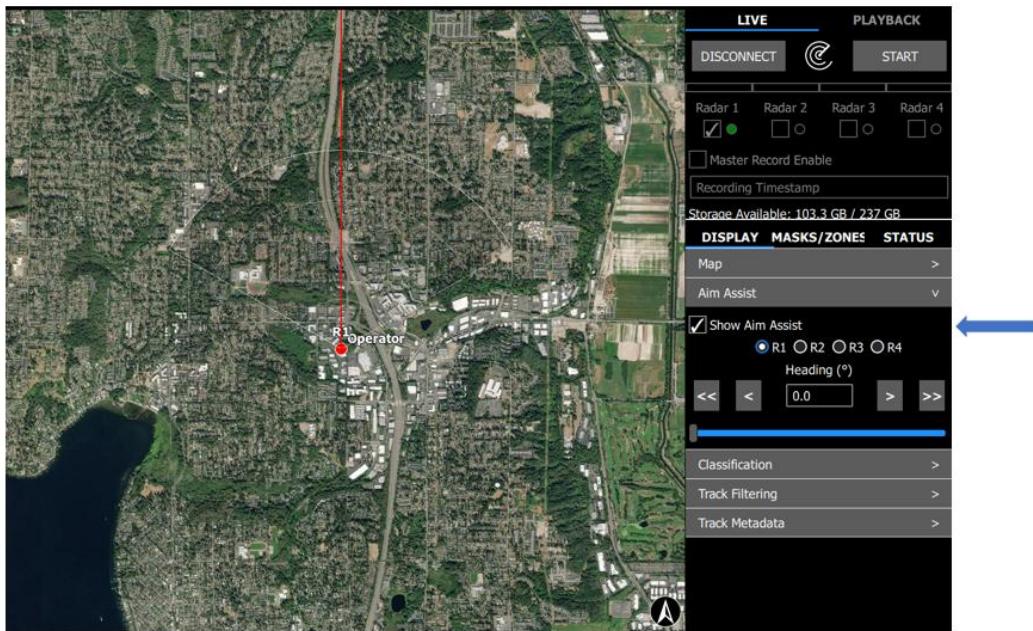


Figure 12 - Aim Assist Heading Alignment Feature

4.7.3 Classification

The classification feature allows users to control the visibility of tracks based on a specified classification declaration threshold. Users can adjust this threshold using a dial, which represents the probability at which a track is classified as belonging to a specific class. By default, all classes are selected to be visible with a class declaration threshold set at **95%**. Tracks with a classification probability below this threshold are labeled as "**Undeclared**." If many tracks appear as "Undeclared," users can lower the class declaration threshold to display more tracks. Users can view the track's classification and its corresponding probability by selecting **p(Class)** under the **Track Metadata** section in RadarUI. This information helps users fine-tune the class declaration threshold for optimal performance.

Tracks will be shown as long as the checkbox for their declared class is checked. If a track is declared as "**Undeclared**" (i.e., it does not meet the declaration threshold), it will still be shown if the "**Undeclared**" checkbox is enabled.

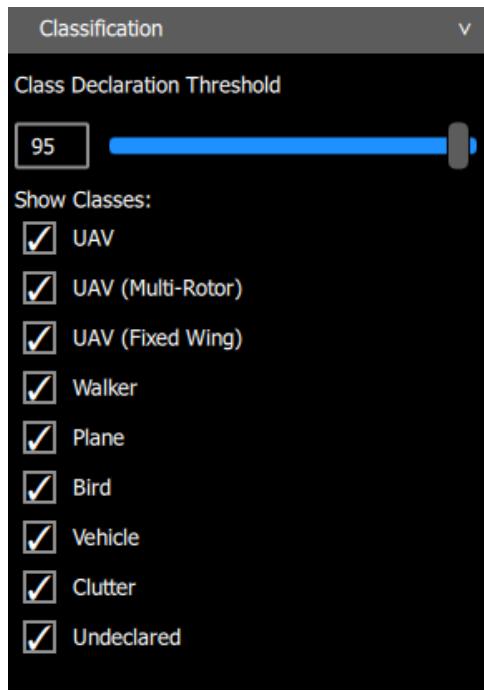


Figure 13 - Classification Controls



Tracks are color-coded by their classification. The colors used are shown in the “Track Class Legend” and can be customized by selecting the respective color box in the legend.

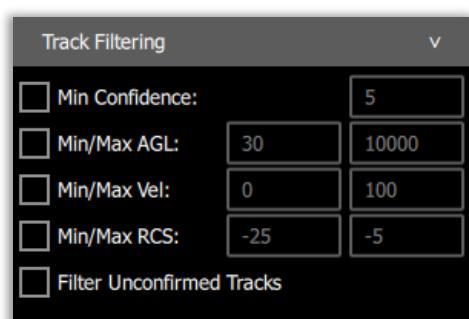
Note:

- EchoShield Radar Software Suite 7.8 support UAVs multi-rotor, UAV fixed-wing, walker, planes, Bird, vehicles, and clutter.
- EchoGuard Radar Software Suite 17.0 and 18.0 supports UAVs, planes, vehicles, walkers, birds and clutter.
- Classification will default to using the binary classifier if the track data does not have a supported multiclass classifier. Tracks will then appear as UAV or Undeclared.

4.7.4 Track Filtering

Track filters help focus on targets of interest by hiding tracks that do not meet the filter criteria. The following filters are available on all Echodyne radars:

- Min Confidence: Metric used for track lifecycle management once tracks are confirmed.
- Min/Max AGL: Minimum and maximum above ground level in meters.
 - On EchoGuard and EchoFlight, “AGL” is the height relative to what’s set as the radar Altitude in the Radar Table (see Section 4.3.2), rather than true AGL.
 - On EchoShield, the AGL filter behaves differently depending on if extended tracks are enabled:



- [Default] With extended tracks enabled, this is true AGL (0m = level with the ground).
The altitude input in the Radar Table must be MSL / orthometric height for AGL to report correctly.
- If extended tracks are NOT enabled, this is not true AGL but instead the altitude above the radar position (0m = level with the radar, not the ground).
- Min/Max Velocity: Minimum and maximum velocity in m/s.
 - Note: The track packet reports the target velocity for that particular time, and the velocity should be considered an instantaneous value. This is not a time-averaged value.
- Min/Max RCS: Minimum and maximum radar cross section, a measure of how detectable a target is by the radar. Higher RCS typically correlates with larger targets.
- Filter Unconfirmed Tracks: When a new detection is first observed by the radar, its ‘track state’ is ‘unconfirmed’. After the detection is observed and measurements are received that exceed the track minimum confirmation threshold, its ‘track state’ is promoted to ‘confirmed’.

4.8 Masks/Zones

The ‘MASKS/ZONES’ sidebar opens the mask editing panel.

4.8.1 Global Masks [EchoGuard and EchoFlight Only]

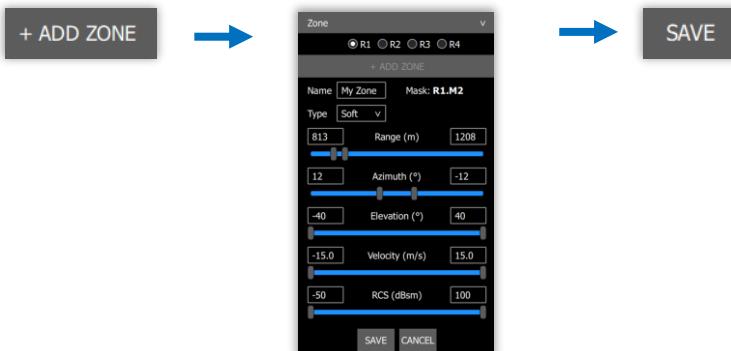
There are three Global masks: Clutter (velocity), RCS (min/max) and AGL. These work similarly to the track filters in the Display tab, except masks edit radar parameters to prevent detections or measurements from being sent to the tracker, whereas filters are applied in RadarUI after tracks are formed and simply hide certain tracks from being output or displayed to the user.

- Clutter Mask Width: This edits the RSP:CLUTTERMASKWIDTH parameter to update the velocity mask width, expressed in number of doppler bins around the zero velocity bin, used to mask out low velocity detections during detection processing. This should typically be left at the default value of 3. See EchoGuard user manual for details.
- RCS Masks: Masks out targets according to radar cross section, which roughly corresponds to target size.
- AGL Masks: Masks out targets according to the target’s relative height above or below the radar.

These masks are EchoGuard and EchoFlight only – although they will show in the UI, they will not have any effect on EchoShield.

4.8.2 Zone Masks

Zone Masks define an enclosed region of three-dimensional (3-D) polar space (Range, Azimuth, Elevation) in which to either mask off tracks or generate an alert when a track enters the zone. To create a new zone, click Add Zone, use the sliders to configure the zone, then click Save.



Mask : Name	Type	Enable
R1.M0 : Alert #1	Alert	<input checked="" type="checkbox"/>
R1.M1 : Alert #2	Alert	<input checked="" type="checkbox"/>

+ ADD MASK

To edit a zone, click on the zone in the box at the bottom.

Mask : Name	Type	Enable
R1.M0 : Alert #1	Alert	<input checked="" type="checkbox"/>
R1.M1 : Alert #2	Alert	<input checked="" type="checkbox"/>
R1.M2 : My Zone	Soft	<input checked="" type="checkbox"/>

Once a zone is created, the type of zone can be edited by clicking the box under “Type”. There are three types total: Two types of zone masks (Hard and Soft), as well as Alert zones.



The zone type can also be set when creating the zone.

4.8.2.1 Hard Zone Mask (EchoGuard and EchoShield)

These zones stop tracks from being formed within the zone under any circumstances. Tracks will not originate within the zone, and tracks originating outside the zone will be dropped once the target enters the zone.

4.8.2.2 Soft Zone Mask (EchoGuard Only)

These zones stop tracks from being formed that originate inside the zone; however, tracks originating outside the zone will continue to exist and be updated as normal as the target enters and leaves the zone.

4.8.2.3 Zone Mask Examples

Since radar scans a large volumetric region, often within the field of view there are regions of spurious movement which can spawn numerous tracks that are not of interest. Examples include busy roads, irrigation heads, HVAC fans on buildings, bird locations such as duck ponds, as well as windblown trees and bushes. To avoid saturating the tracker with uninteresting targets, the user can place up to 32 unique and overlapping zone masks throughout the FOV.

By default, newly created masks are initially located in the middle of the FOV with all elevation, velocities and RCS values selected. Masks can be enabled/disabled in the UI to assess their affect and dynamically edited while the radar is operating.



Figure 14 - Settings Sidebar: MASK Controls

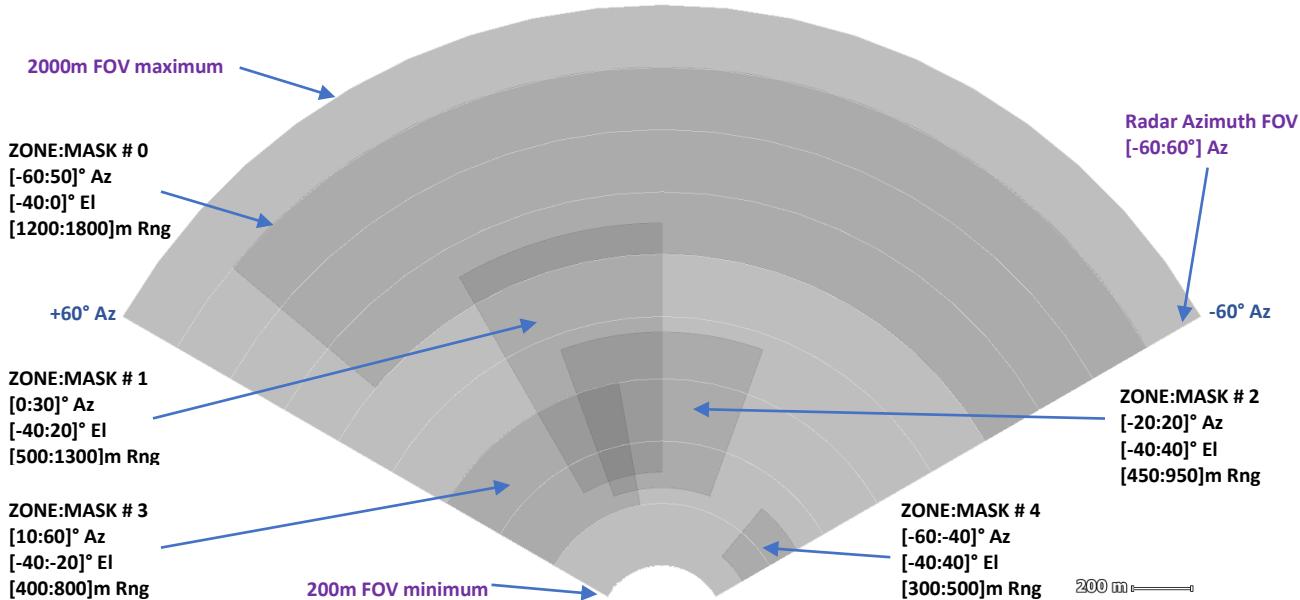


Figure 15 – Example overlapping Zone Masks within Radar FOV, showing Range vs Azimuth Plan View

Notes on Zone Mask Behavior:

Configuring 3-D zone masks can be a complex task. Below are additional details for optimum performance.

1. Up to 32 unique zone masks (0-31 slots) may be specified.
2. Zone masks may overlap in 3-D space without issue, as the comparison engine checks measurements for inclusion within each zone sequentially starting at zone mask 0.
 - a. For optimal processing efficiency, overlapping zone masks should be specified starting at slot 0 in order from largest to smallest so each measurement is rejected with the fewest comparison operations.
3. Negative Azimuth from the radar POV is to the right. Negative Elevation is intuitively below radar.
4. Soft Zone Mask (light gray) = Tracks can propagate through, but not originate in a region
5. Hard Zone Mask (dark gray) = Kills all tracks that enter a defined region
6. All zone mask parameters are currently integers except vRadial which is floating point.
 - a. One can specify zone masks in steps smaller than the beam step as they operate on measurements which are interpolated detections across angles and ranges.
7. Zone Mask measurement velocity is the radial Doppler velocity component as detected by the radar within the maximum unambiguous velocity (MUV). This can be a complicated concept to initially grasp for new users. Remember that only objects moving at least partially toward or away from the radar have a detectable Doppler velocity component.
 - a. Consequently, the Zone Mask velocity range spans the -MUV to the +MUV of the radar with negative velocities approaching and positive values departing from the radar location. See the radar user manual for details.
8. The RCS range spans -50 dBsm (insect) to +100 dBsm (brighter than any conceivable radar target).

4.8.3 Alert Zones

For EchoGuard and EchoShield radars, Alert Zones create a zone where an alert will be generated if a target originates within or pass into the zone. Alert Zones show as yellow in the UI, and turn red when a track enters the zone. An audible alert chime also sounds when an alert is triggered.

If an Alert Zone overlaps with a Zone Mask, a pop-up message will appear to alert the user of this issue. An alert will not be generated in the overlap region if the Zone Mask kills the track and/or prevents a track from forming.

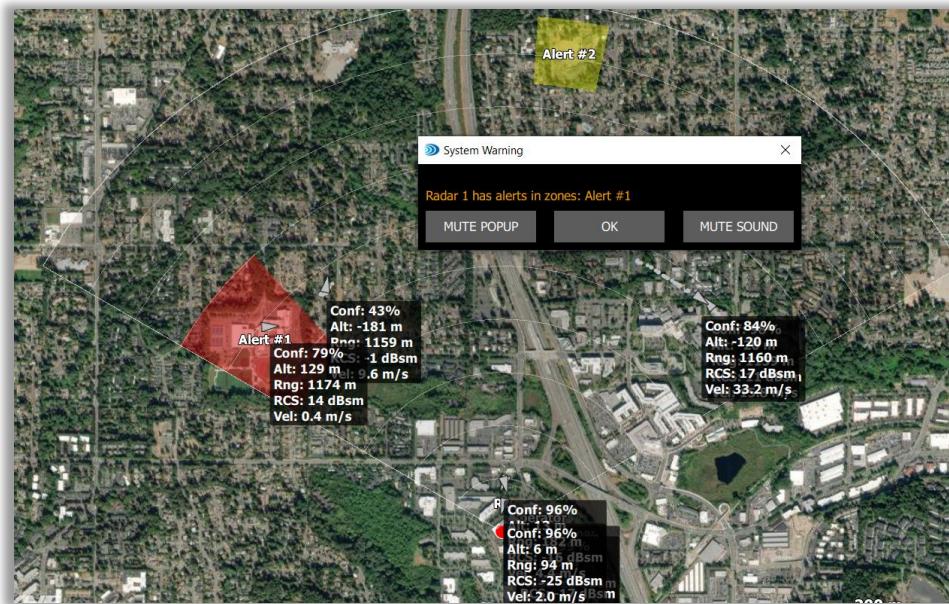


Figure 16 - Alert Zone turns red and generates a popup and audible chime when target enters the zone

After all targets leave the alert zone, the alert will be gone, but the popup box will remain so the user knows an alert occurred even if that alert is no longer active.

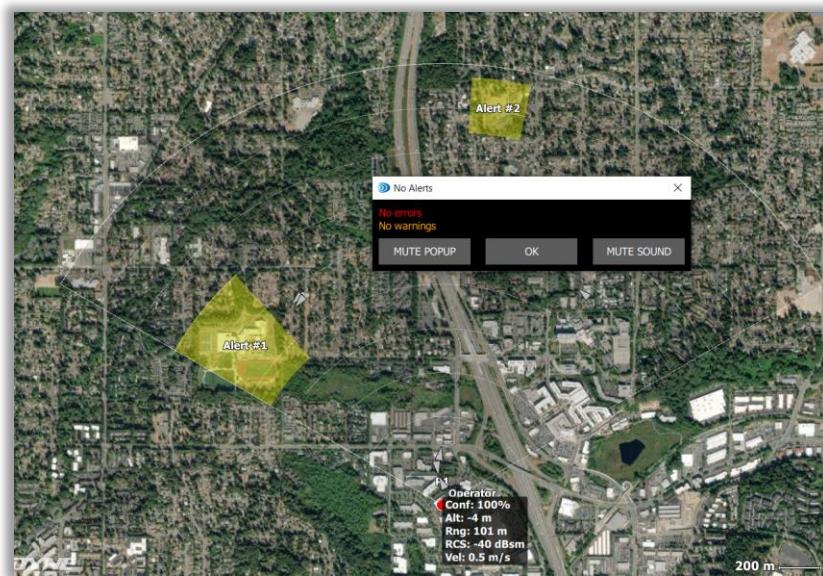
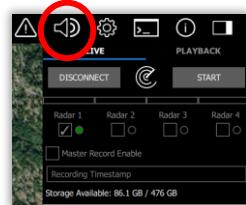


Figure 17 - Popup remains after target leaves the alert zone



To mute the visual alert, click the MUTE POPUP. To mute the audio alert, click MUTE SOUND or the speaker icon in the top right.

4.8.4 EchoShield Masks (EchoShield Only)

Zone Masks define an enclosed region of three-dimensional (3-D) spatial space (list of latitude, longitude points defining a polygon with an optional maximum height above ground) in which to mask off tracks. To create a new polygon zone mask, click Add Mask, use the mouse right button to create latitude/longitude points, click back on the first point to close the polygon shape, enter an AGL if desired, then click Save. Before closing the shape, you can click Undo to remove the last point added.



To edit a zone, click on the zone in the box at the top. When editing you can reshape the polygon (click on a marker and moving it) and add/modify the label and AGL.

Name	AGL (m)	Enable
HS	∞	<input checked="" type="checkbox"/>
Park	55.0	<input checked="" type="checkbox"/>

A polygon zone mask cannot be self-intersecting. If a polygon zone mask is self-intersecting, attempting to save the mask will pop up an error and not be saved.

These (soft) zones stop tracks from being formed that originate inside the zone; however, tracks originating outside the zone will continue to exist and be updated as normal as the target enters and leaves the zone.

4.8.4.1 EchoShield Zone Mask Examples

Since radar scans a large volumetric region, often within the field of view there are regions of spurious movement which can spawn numerous tracks that are not of interest. Examples include busy roads, irrigation

heads, HVAC fans on buildings, bird locations such as duck ponds, as well as windblown trees and bushes. To avoid saturating the tracker with uninteresting targets, the user can place up to 32 unique and overlapping spatial zone masks.

Masks can be enabled/disabled in the UI to assess their affect and dynamically edited while the radar is operating.

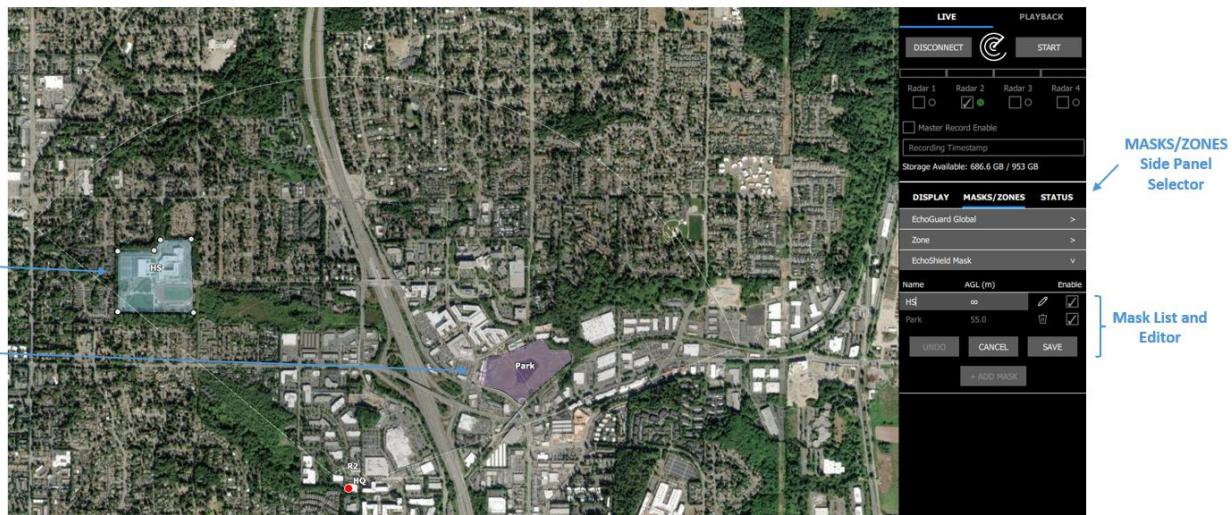


Figure 18 - Settings Sidebar: EchoShield MASK Controls

Notes on Zone Mask Behavior:

Below are additional details for optimum performance.

1. Up to 32 unique spatial polygon zone masks (0-31 slots) may be specified.
2. All enabled masks are programmed across all connected EchoShield radars.
3. There must be between 3 – 32 <latitude, longitude> pairs (i.e., polygon vertices).
4. Zone masks may overlap without issue.
5. A polygon cannot be self-intersecting.
6. A Zone Mask is Soft – Tracks can propagate through, but not originate in a region.
7. Optional maximum height above ground setting. Defines the maximum height above ground for which the mask extends to.
 - a. If not specified, infinity is assumed (light gray) = Maximum height is used.
 - b. If AGL is provided (light purple) = The specified height is used.

4.9 Map & Data Display

The focal point of RadarUI is a high fidelity geo-located presentation of tracks and their associated metadata on an underlying map. This enables the user to rapidly associate radar tracks to real-world objects, optimize and troubleshoot radar settings and track specific targets of interest. Multiple radars (up to 4) can be simultaneously displayed in real-time with overlapping fields of view.

Optimization of the radar settings for optimal tracking is best covered by the radar User Manuals and various application notes. Below is a high-level summary of the key features of the map display.

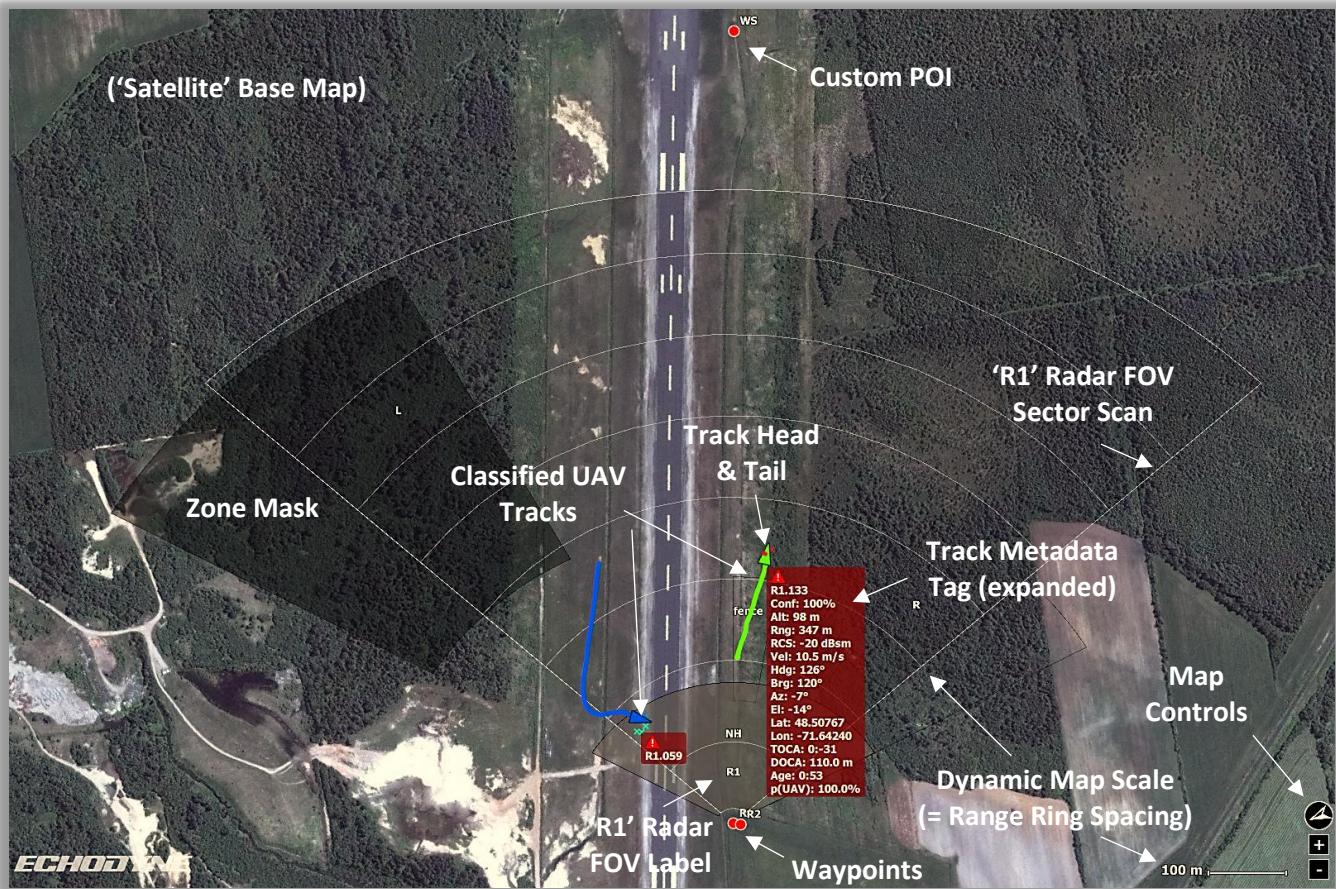


Figure 19 - Map Display Key Features & Controls

4.9.1 Key Map Controls

- **Pan Map** – Click on map and drag with either mouse or finger (touchscreen).
- **Zoom Map** – Middle mouse button scroll, tap + / - controls at bottom right or pinch zoom (touchscreen).
- **Rotate Map** – Shift + middle mouse button scroll or two-finger rotate (touchscreen). Notice compass needle at button right rotates to indicate true North.
 - **North Up** – Click/touch on compass needle to snap to ‘North-up’ view.
 - **Radar Up** – Double-click on the radar FOV label (e.g. ‘R1’) to snap to ‘Radar-up’ view. Multiple radars supported.
- **Base Map Toggle** – Use Setting Sidebar control to toggle between satellite and street map views.

4.9.2 Map Sources & Tile Caching

At present, the underlying base map options are satellite and street views provided by ESRI™ (Environmental Systems Research Institute, Inc.) and Mapbox™. While the primary intended RadarUI usage assumes an active internet connection, limited offline caching of map tiles is possible. As one manipulates the map via panning & zooming, the map API triggers new tiles to load in the /bin/tiles folder up to the 100 MB total tile file size limit. To load

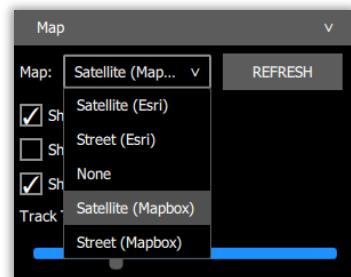


Figure 20 - Map tile options

various ‘map sets’ for offline field use, one can copy and paste into the /bin/tiles folder various tile file collections.

4.9.3 Track ‘Metadata Tag’ Toggling

Tracks are initially presented as color-coded arrows reflecting target true heading and a user-adjustable (0 to 60 second) track tail ‘history’ generated with color-coordinated breadcrumbs. The breadcrumb dots represent individual track updates and are plotted at the radar track update rate, typically 5-10 Hz.

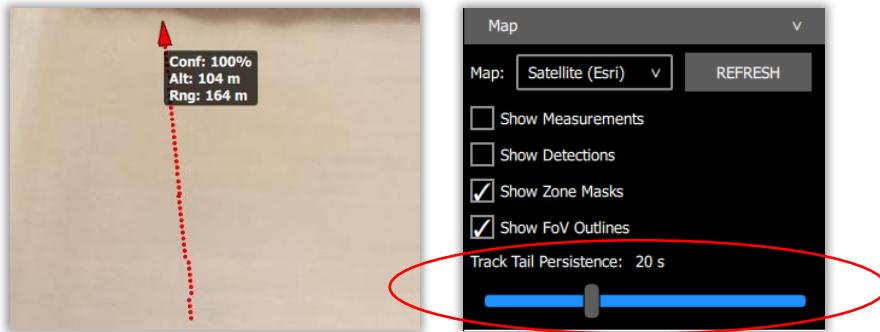
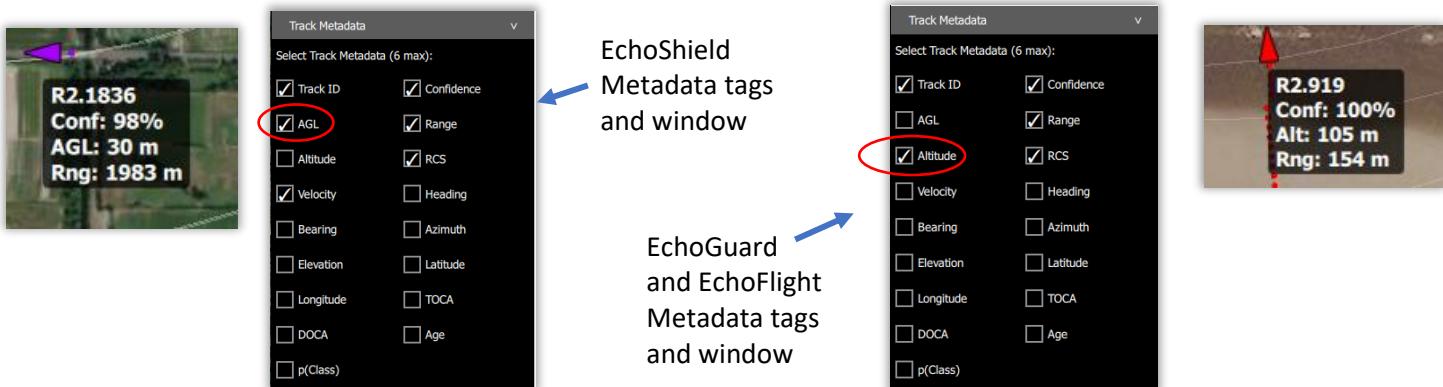


Figure 21 – Track Tail Persistence Setting

Associated with each arrow is a moving metadata window tag which initially shows the user-selected subset (up to 6 fields) of track information. On EchoGuard and EchoFlight, “Altitude” is the height relative to what’s set as the radar Altitude in the Radar Table (see Section 4.3.2). For example, if an EchoGuard is set to an Altitude of 7m in the Radar Table, and a target is flying 100m above the ground, the metadata “Altitude” (designated “Alt” in the metadata window) will show as 93m.

On EchoShield an additional “AGL” box will be shown. This field is true Above Ground Level when extended tracks are on, and height above radar when extended tracks are off. The metadata window displays this as “AGL”. **The “Altitude” box can be selected, but it will not have any effect on EchoShield.**



The metadata window can be selected via mouse click to show all track metadata. A third click will suppress the metadata window leaving just the arrow. This allows the user to easily interrogate and follow tracks of interest while muting uninteresting ones.

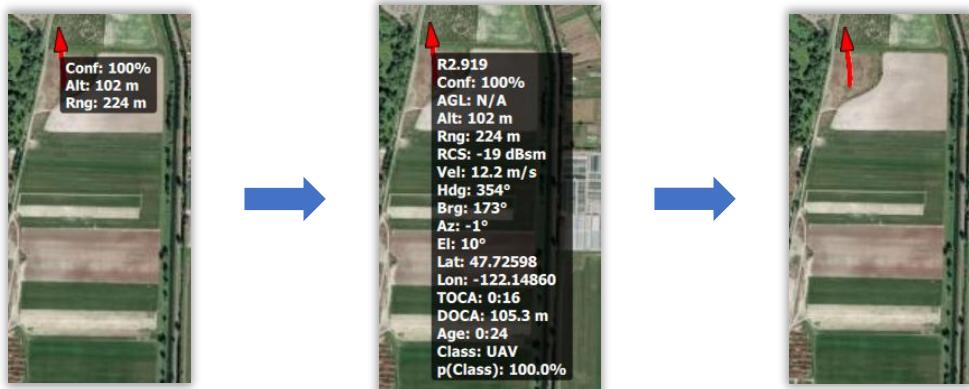


Figure 22 – Cycling through the three track metadata states (user default → maximized → hidden)

When the track ‘dies’ and the radar ceases to update the track, the metadata data window will disappear, but the arrow tip will persist (last known heading) as set by the Track Tail Persistence slider. The arrow can be selected (mouse click or finger touch) to bring back the metadata window and query the ‘last known location’ of expired track. After the track tail fully expires, the arrow will disappear and no longer be accessible.

Track Metadata Notes:

1. Track metadata (e.g., altitude, bearing, latitude, longitude) requires accurate configuration file settings for correct data presentation and orientation to the underlying maps.
2. On EchoGuard or EchoFlight, if radar altitude is set to its mounting height above local ground in meters, radar tracks will be reported relative to ground. If radar altitude is set to local elevation relative to sea level, radar tracks will be reported in MSL. Operator’s choice depending on the use case.
3. With long track persistence times (e.g., 60 seconds) the display can become cluttered with dead track graphics. Use the ‘REFRESH’ button to clear the screen of all but active ‘live’ tracks. Shortening the track persistence can also help for busy scenes.

4.10 Radar Data Playback

RadarUI can playback recorded raw radar packet data. By selecting the ‘PLAYBACK’ sub-window, a new set of controls is available and the configuration gear icon adds a playback triangle. Selecting the configuration gear icon brings up the Playback data selection window.

If the radar data was recorded via the UI, then a timestamped *RadarConfig_year-mo-dayThrminsec.json* file is saved that captures the exact configuration of the radar. Use the ‘Import’ button to open a file selector and navigate to where this file was saved in the */radarui_data* folder within the user defined recording directory. The raw radar data file location will be automatically located based on matching the folder timestamp.

If the raw data was not recorded via the UI application, it is still possible to playback a specific radar data folder. In this case, the ‘USE CURRENT’ button will copy the ‘LIVE’ mode radar orientation and allow you to select a specific folder location for playback.

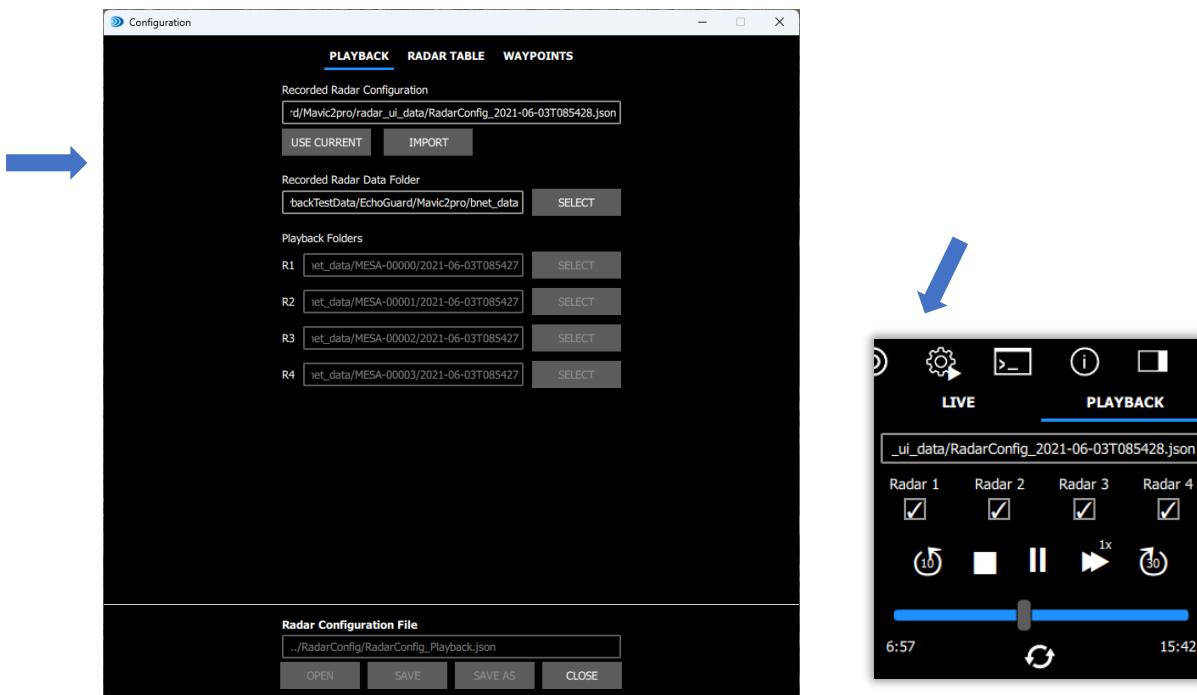


Figure 23 - Radar Data Playback Configuration Screen & Controls

Once the data is located, the playback controls are very similar to any video display application. One can play, stop, pause, speed up (1x to 16x real-time), jump forward (30 sec) or backward (10 sec) and use the slider to jump to a specific time location.

During playback, the UTC time clock at top left will reflect the time the radar data was recorded from reading the radar data packet headers.

5 FAQ & Troubleshooting

- **My SysConfig.json and/or RadarConfig.json got changed/overwritten!**

If the JSON file is detected as malformed, RadarUI will overwrite this file to defaults. It saves a timestamped back-up in the local folder location for user recovery.

- **Application is slow to start up**

If the location specified by Waypoint #1 (map start-up location) is in an area with no map tiles exist, the application may be slow in loading while it tries to retrieve new tiles over a slow internet connection.

- **I can't connect to the radar (red dot next to check box)**

Confirm power supply, cabling and that the Ethernet IP address is correctly in the UI. Try the EchoGuard BNET CLI application or EchoShield SDK for basic troubleshooting or simply ping the radar at its IP address to confirm connectivity.

- **My radar is missing data (yellow dot next to check box)**

It is possible to operate the radar via the RadarUI and have another program 'steal' a data port such as the track packets. In this case, the UI will still maintain connection with the command port but will detect one or more missing data streams and change the radar connection indicator to yellow. The corresponding missing data will not be plotting on the map. If/when the port connection is recovered, the indicator will revert to green.

- **I'd like to run the RadarUI application in parallel with my own C2 software to compare results in real-time.**

This is a common request and we would recommend a low cost, 3rd party software called TCP Splitter (<https://www.agisoft.com/tcp-splitter/>) which can mirror TCP/IP ports and provide exactly this functionality. If interested, we have an application note that describes how to configure this. Additionally, **Monitor Mode** allows for parallel operation with the 3rd party C2 controlling the radar(s).

- **I do not see a map underlay or map tiles are missing a different 'zoom' levels.**

Map tiles must be downloaded from the map provider for the location of interest and require an internet connection. For remote field tests, it is advised to pre-cache the tiles of interest by previewing the intended area of operation at various zoom levels. Missing tiles will be blank.

- **The display is super busy with too many tracks!**

Time to start tuning the radar for the operational environment! Consult the respective radar User Manual for commands to control scanning field of view and various masking knobs (RCS, Velocity/Clutter, Range, etc.). Or utilize the track filtering options in the UI to help clean the scene. When in doubt, contact support@echodyne.com for sample set-ups and recommendations.

- **My question or issue isn't listed here. Help!**

Help us add things here you find non-intuitive or non-functional. Email to support@echodyne.com.