



# JS8Spotter

Search Term Traffic Scanner  
And Other Utilities for Use With JS8Call

**License:** (c) 2024 Joe Lyman. Permission is granted to use and adapt the contents of this document as needed, with the exception of any personally identifiable private information such as license identifiers, names, organization names or membership, email addresses, and phone numbers, which may not be redistributed. You may remove this license notice on adapted versions or extracts only if all personally identifiable private information, as described above, is removed as well. This is not an official document of any formal organization.

**Table of Contents**

Overview.....4

Prerequisites.....4

    Windows users.....4

    Linux Users.....4

    Upgrading From a Previous Version.....5

Software Settings.....6

    In JS8Call:.....6

    In JS8Spotter:.....7

    Miscellaneous JS8Spotter Settings:.....7

QuickStart.....8

Software Features.....9

    1. Profile-Based Activity Search.....9

        Creating, Editing, or Removing a Profile.....10

        Background Scan.....10

        Search Terms.....10

        Adding and Removing Search Terms.....10

        Importing and Exporting Search Terms.....11

        Activity Matching.....11

        Quick Filter.....11

    2. Expect Subsystem.....12

        The Expect Window.....12

        Adding an Expect Statement.....13

        Edit, Remove, or View an Expect Statement.....13

        Importing and Exporting Expect Statements.....13

        Triggering Expect Responses.....13

        Sending an Expect Response Manually.....14

        Expect Operating Tips.....15

        Pausing the Expect System.....15

        Expect Blocklist.....15

    3. Multiple Choice / Micro Forms.....16

        Creating Form Templates.....16

        Filling Out and Posting Forms.....18

        Transmitting a Form Over the Air.....19

        Form Responses.....19

        Response Gateway.....20

        MCForms Datecode.....20

    4. Additional Features.....21

JS8Spotter de KF7MIX, Companion Application for JS8Call de KN4CRD

APRS Commands.....21  
Simple Offline Maps.....21  
Visualize Waterfall.....22  
CommStat Compatible Features.....22  
Simple Database Search.....22  
Database Trim.....22  
Getting Additional Help.....23

## Overview

*JS8Spotter* is an API (application programming interface) based companion application for JS8Call. It communicates with your running JS8Call instance over TCP (transmission control protocol; on your local machine or optionally over a network). *JS8Spotter* is not a standalone application. It requires a running instance of JS8Call to operate.

*JS8Spotter* provides several useful features in a single Python file (with supporting theme, form, and image files). Information gathered by *JS8Spotter* is stored in an SQLite3 database, and can be accessed through freely available third-party software.

At its base *JS8Spotter* **provides a way to apply automatic keyword searches to incoming JS8Call traffic, so that operators can more easily store and track traffic that interests them.** This is the primary function of *JS8Spotter*. Additional features include expanded auto-responders, micro forms, APRS commands, as well as map and waterfall visualizations.

## Prerequisites

In order to use *JS8Spotter*, you must have a fully functional JS8Call setup. If you're unfamiliar with JS8Call, you will need to start there first. There are **no** radio, sound card, or other hardware configuration requirements related to *JS8Spotter*; all of those concerns are handled in JS8Call.

It is highly recommended that operators looking to use *JS8Spotter* be very comfortable with JS8Call. Learning JS8Call first will help operators understand the need for and value of *JS8Spotter*. Configuration issues having to do with the operation of JS8Call itself are outside the scope of this document.

## Windows users

If you're using the binary build, everything should be setup for you to run the program using the following instructions. Visual C++ Redist package may be required. If you're using the source, you may need to scan through the Linux Users section below to get hints about configuring Python3 and the libraries.

Download the latest Windows build from <https://kf7mix.com/js8spotter.html> and save the .ZIP file where you would like to unpack it. Locate the file in File Explorer, right-click and Extract All. A new folder should be created for *JS8Spotter*. Enter the folder and double-click the application icon to start the program.

## Linux Users

You will require a functional Python3 configuration with Tkinter and Pillow ImageTk libraries installed. On most distros, this is a simple matter of installing the needed packages using pip or your package manager (apt, etc.). On a default Linux Mint 21.1 installation (where Python3 is included), for example, the following command should suffice to setup a basic environment:

```
"sudo apt install python3-tk python3-pil python3-pil.imagetk"
```

## Upgrading From a Previous Version

If you're upgrading from a previous version of JS8Spotter, please note the following:

- JS8Spotter runs in-place, which means that all of the files it uses are included in the same directory
- You may have more than one copy of JS8Spotter on a single system, of different versions
- The js8spotter.db database file contains all of your user settings, profiles, and gathered activity
- If you have custom WAV notification sounds, or custom forms, you will need to copy those manually into the new installation after the database is updated

To upgrade, you'll need to complete the following actions:

- Make a backup of your previous version js8spotter.db file
- Download and unpack the latest version of JS8Spotter; Do not overwrite your existing copy
- Place your old js8spotter.db file in with the newest version, overwriting the newer one
- Read the included .txt instructions, then run the dbtool (either with the exe or .py file)
- After your database file is updated, test the newest version to make sure it works
- After everything is working, feel free to replace your old version or update your links

Because JS8Spotter runs in-place, with no additional files installed anywhere on your system, you can download, upgrade, and test without losing your working copy. Please ensure that the new version works for you before doing anything with your existing copy! There is no reason to lose your working version while upgrading. Follow the steps above carefully to ensure you can upgrade, and reach out to me if there are issues.

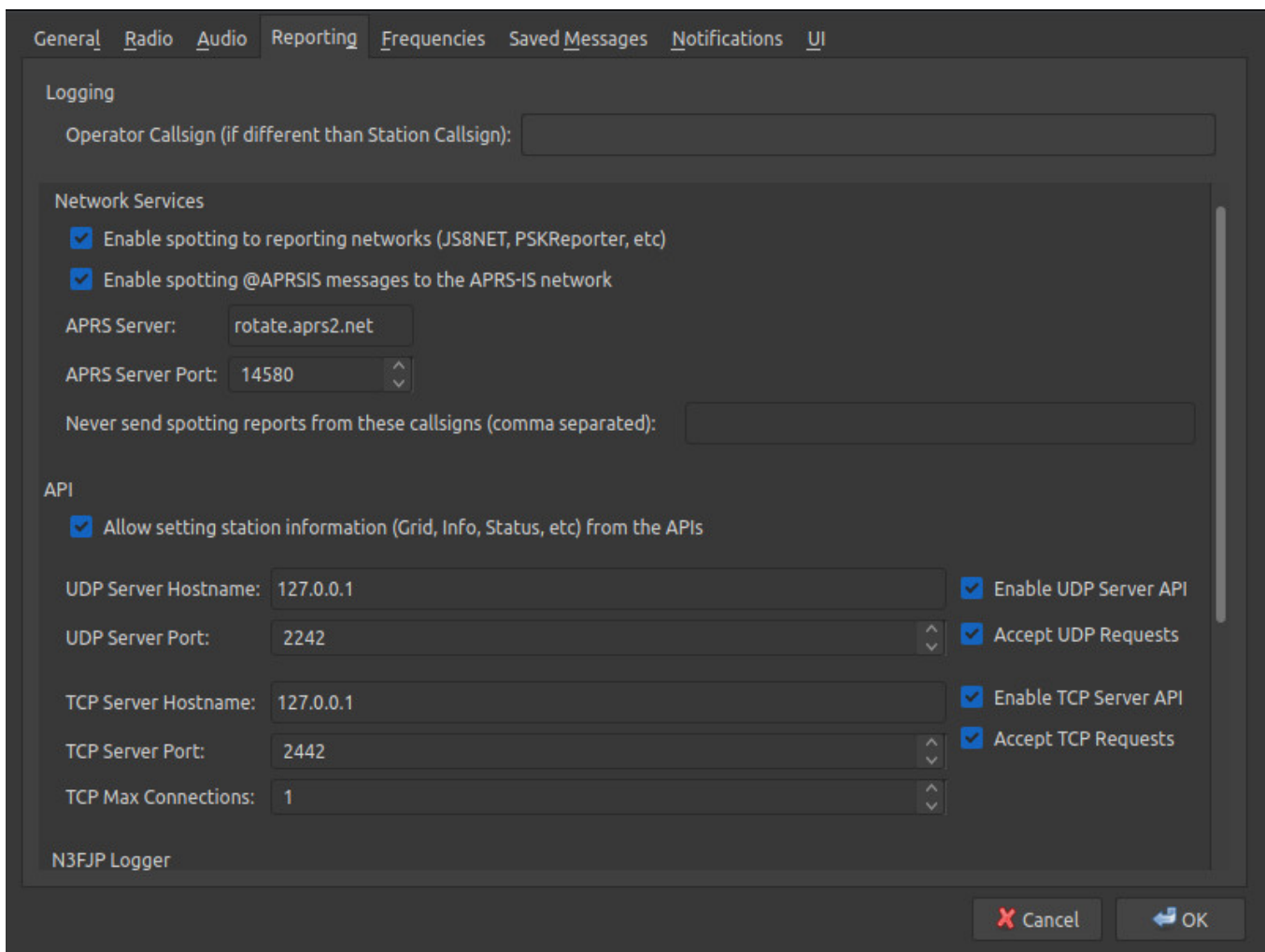
**ALWAYS BACKUP YOUR JS8SPOTTER.DB FILE**  
**AND ANY CUSTOMIZED FORM OR WAV FILES**  
**BEFORE TRYING A NEW VERSION**

## Software Settings

Once you have a functional JS8Call setup and Python3 environment, you'll need to verify that your settings are correct so that your JS8Call API is accessible in *JS8Spotter*. Sane default settings are included, but it is up to you to make sure that everything is set and matching in both applications.

### In JS8Call:

- In JS8Call, load the settings by pressing "F2", or by using the "File>Settings" menu item
- Visit the "Reporting" tab
- Make sure the "Allow setting station information..." check box is enabled
- Make sure the "Enable TCP Server API" check box is enabled
- Make sure the "Accept TCP Requests" check box is enabled
- Take note of the "TCP Server Hostname" and "TCP Server Port" values



## In JS8Spotter:

- Ideally, make sure JS8Call is running before running *JS8Spotter*
- In *JS8Spotter*, load the settings by using the "File>Settings" menu item
- Enter your Callsign and GRID (4 character)
- Ensure that the "IP Address" and "TCP Port" in *JS8Spotter* match the settings in JS8Call
- If you use the CommStat features, enter your group name, otherwise leave blank

Please check that these settings match in JS8Call.

Your Callsign	KF7MIX
Your GRID	EM48
IP Address (127.0.0.1)	127.0.0.1
TCP Port (2442)	2442
CommStat Group (or blank, or *)	

Save Cancel

## Miscellaneous JS8Spotter Settings:

The following optional settings in *JS8Spotter* will help you control how you wish to view and operate the software.

- "View>Hide Heartbeats" will let you hide heartbeat traffic from the "Matched Activity" pane. You may also Hide/Show RX.SPOT, RX.ACTIVITY and RX.DIRECTED items
- "View>Dark Theme" will let you toggle between a light and dark software theme
- "View>Highlight Range" will temporarily highlight new activity in the "Search Terms" pane
- "Tools>Expect – Pause Replies" menu item will disable all Expect auto-replies
- "Tools>Disable Sounds" will temporarily disable all profile notification sounds
- If you plan to use Expect or APRS commands to transmit from JS8Spotter, the Auto-Respond setting in JS8Call must be enabled

## QuickStart

This section is primarily for those that prefer hands-on learning. Each of the primary features of *JS8Spotter* are described in very high-level detail below. There is enough here to get you started. For additional information, see each of the corresponding sections in this manual.

- **Profile-Based Activity Search:** Primary function of *JS8Spotter*. Create profiles (i.e. "Contacts", "Chess", "Weather"). Enable "Background Scan" and they'll scan even when not open. In each profile, create search terms to scan for (i.e "KF7MIX", "HURRICANE"). The same search terms may appear in multiple profiles. As traffic comes in through the JS8Call API, *JS8Spotter* will look for matches and save matched traffic in the database.
- **Expect Subsystem:** This is a user-defined auto-responder. Your station will "expect" some text, and "respond" with text you provide. It will only "reply" to stations you allow, and it will only auto-reply the number of times you permit.
- **MCForms:** Micro Forms aka Multiple Choice Forms. Look in the forms folder for example forms. Create your own forms if desired, and publish them on kf7mix.com or on your own site. Fill in forms and post them to the Expect Subsystem. Your station will automatically retrieve forms seen on the air and store them.
- **APRS Commands:** *JS8Spotter* includes shortcuts for sending APRS commands through JS8Call. Simply fill in the boxes and send.
- **Simple Offline Maps:** Maps show the approximate location of stations that are reporting their GRID. Helps with understanding propagation. North America, Europe, Australia, and Indonesia are included, but you may create your own maps for anywhere in the world.
- **Visualize Waterfall:** A static visualization of the JS8Call waterfall with signal location and size represented for a period of time from 5min to 24hrs. A good way to see which parts of the waterfall are not as widely used.

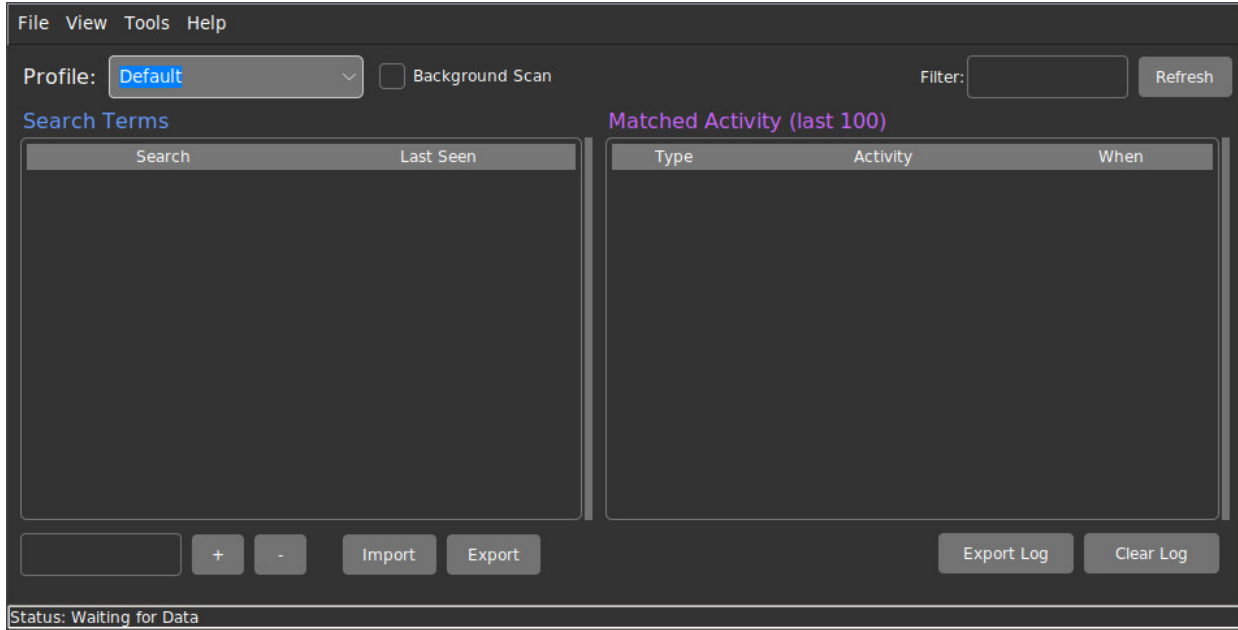


# Software Features

Each of *JS8Spotter's* four main features are covered in detail below.

## 1. Profile-Based Activity Search

This is the main function of *JS8Spotter*. When you first open the application, the main window display has two panes: On the left, the "Search Terms" pane; and on the right, the "Matched Activity" pane.



You may wish to create multiple profiles, to organize the search terms and traffic you wish to track. For example, if you wish to track your contacts/friends as well as weather events, you might have profiles and search terms like these fictional ones:

<b>Profile:</b> Contacts	<b>Profile:</b> Weather
<b>Search Terms:</b> KF7MIX ABC1DE @JS8CHESS JOHN SMITH	<b>Search Terms:</b> WX HURRICANE STORM TRACK FLOOD WATCH

Users may create as many profiles and search terms as they like. The same search terms may appear in multiple profiles, at the user's discretion.

## Creating, Editing, or Removing a Profile

To create a profile, you simply use the "File>New Profile" menu item. Enter the name of your profile in the box, and create it. To edit a profile name, use the "File>Edit Profile" menu item, edit the name and save it. You may set a notification WAV sound in the "Edit Profile" screen, which is a single WAV filename located in the spotter folder.

To remove a profile, first make sure you have that profile selected by using the "File>Switch Profile" menu item, or by using the select box in the main window. Next, use the "File>Remove Profile" menu item. Removing a profile will remove all the associated search terms and traffic.

## Background Scan

In the main window, next to the profile select box, is a check box for "Background Scan".

- When this box is  **unchecked**, search terms in this profile will only be scanned when this profile is actively selected and displayed in the main window
- When this box is  **checked**, search terms in this profile will be scanned in the background, even if you are currently viewing a different profile.

## Search Terms

Search terms may be single or multiple words, using any standard ASCII characters you wish to use. All search terms will be converted to capital letters.

Search terms are matched exactly only. If you wish to match narrowly, use exact terms such as a call sign or a full word (i.e. "KF7MIX", "THUNDER STORM", "YELLOW CARS"). If you wish to match broadly, use partial terms (i.e. "KF7", "STORM", "CAR").

Use multiple variations of search terms and/or search concepts if needed (i.e. "WX", "WEATHER")

## Adding and Removing Search Terms

There are two ways to add search terms to a profile. Make sure the correct profile is selected, then do one of the following:

1. Enter a search term into the box below the left "Search Terms" pane. Press <Enter> or click "+" to add the search term to your list.
2. Click the "Import" button beneath the left "Search Terms" pane. In the space provided, type or paste your search terms, one search term per line, then click "Add Batch".

To remove search terms from a profile, use one of the two following methods. You will be prompted before the search terms are removed:

1. Highlight the search term and click the "-" button beneath the left "Search Terms" pane.
2. Use <Shift> and/or <Ctrl> to click and highlight multiple search terms, then click the "-" button beneath the left "Search Terms" pane.

## Importing and Exporting Search Terms

Importing is equivalent to adding, but remember that what you import may be typed or provided from another source. For example, if multiple users of a group wish to have a profile that uses the same terms (i.e. a JS8Chess group, an emcomm group, or a Skywarn group), they could share those search terms via email or a text file on a thumb drive. Each user could "Import" the same set of data into *JS8Spotter*.

To export search terms, select the profile you wish to export from, and click the "Export" button beneath the left "Search Terms" pane in the main window. You may copy or save the search terms for later use.

The Search Terms Import and Export features both operate with blocks of text, with one search term per line.

## Activity Matching

The right "Matched Activity" pane in the main shows the latest 100 matches for the selected profile's search terms. Additional matched activity is stored in the database. As traffic comes in to JS8Call, it is provided to *JS8Spotter* through the API over TCP. *JS8Spotter* examines the incoming traffic and attempts to match on search terms in the active profile and in any profiles that have Background Scan enabled. When a match is found, the activity is saved in the database.

To view details on matched activity, double-click an entry in the "Matched Activity" pane. Right-click on a highlighted entry to copy the report contents to your copy/paste buffer.

If you wish to see additional saved activity, locate the associated search term in the left "Search Terms" pane, and double-click it. This will bring up all matched activity for that search term in the selected profile. Double-clicking or right-clicking entries in this view will bring up activity reports or copy them to your copy/paste buffer respectively.

Traffic need not be directed at your station to be recorded. *JS8Spotter* will keep track of any traffic containing the search terms you provide.

As times, JS8Call provides the same traffic to the API multiple times and/or in multiple ways. This may result in duplicate matches in *JS8Spotter*.

## Quick Filter

Above the "Matched Activity" pane you'll find a quick filter box. By entering text in this box, you can filter the results below to contain only activity matching the text. Press <Enter> to immediately filter, or press the Refresh button; or, simply wait for new activity to load and be filtered automatically. To clear the filter, simply erase the text or press <Esc> to clear the box completely.

## 2. Expect Subsystem

The "Expect" subsystem provides a method of expanding JS8Call's user-configurable auto-response ability. The built-in user field commands in JS8Call are as follows:

- **"INFO?"** - This provides station information that you've saved in JS8Call's settings. People usually use this line for rig info, but it may be used for anything.
- **"STATUS?"** - This provides status information that you've saved in JS8Call's settings. People sometimes use this to relay idle time or JS8Call program version information.
- **"GRID?"** - This provides your grid as saved in the JS8Call settings.

As all JS8Call users should be aware, if you send any one of these commands in the blind to any other station with autoreply enabled, that station will immediately respond with the stored text. Expect works the same way, but with more flexibility.

With Expect, you can create your own commands and responses (collectively called "statements"), and control the Callsigns and/or groups that are allowed to trigger them. You also have a log of when they were accessed, and the ability to disable them after a fixed number of transmissions if desired.

**All Expect statements are triggered using the "E? " prefix.** See "Triggering an Expect Response" below.

You should be aware that Expect does not include any statements by default. This means that other stations have no way of knowing that you are running Expect, or what commands are available, unless you choose to expose that information using an Expect statement. Expect relies on operators working together in groups in order to make the feature useful.

### The Expect Window

The Expect window provides the interface to add, edit, remove, import/export, and view Expect statement.

Expect	Response	Allowed Calls/Groups	Sent To	Count	Created
F1103	F1103 N143 ONE 10M JS8 QSO ALBERT *	*		0/∞	2024-01-24
WX	40-51F THRU 1/31	*		18/∞	2024-01-22
F1100	F1100 7234442137 JOE #IJKL *	*		7/∞	2024-01-18
EMAIL	JOE@PINPOINTAZ.COM	*		1/∞	2024-01-14
CMD5	RIG,ANT,WX,QTH,URL,EMAIL,WX,GRP,M *	*		41/∞	2024-01-14
GRP		*		7/∞	2024-01-14
MCF	FORMS STORED: F1100 F1101 F1103 F1104 *	*		26/∞	2024-01-14
ANT	EFHW-8010 30FT	*		14/∞	2024-01-12
RIG	FTDX10 10W PO	*		19/∞	2023-12-28
QTH	FULTON, MO	*		0/∞	2023-12-11
F1101	F1101 40352A231 #KOPY *	*		7/∞	2023-11-15
F1304	F1304 11411111 TEST #EHKU *	*		0/∞	2023-05-08
F1301	F1301 143132111 #COIC *,@	*		0/∞	2023-04-08
URL	KF7MIX.COM	*		5/∞	2023-02-24

Text to Expect (6):  Text to Respond With:  Allowed Callsigns/Groups:  Max Replies:

## Adding an Expect Statement

To add an Expect statement, open the Expect system and fill in the fields at the bottom of the window:

- **"Text to Expect"** – A single word of 6 characters or less; A-Z, 0-9, and "!" allowed, no spaces. Do not include the "E? " prefix here, it is always assumed
- **"Text to Respond With"** – Transmitted when an allowed station or group sends the "Text to Expect"
- **"Allowed Callsigns/Groups"** – List of callsigns or groups that are allowed to trigger this response. It can be the "\*" character for anyone, or a list of comma-separated specific callsigns and/or groups. (i.e. "\*" or "\*,@JS8CHESS" or "KF7MIX" or "KF7MIX,AB1CD,EF2GH" or "KF7MIX,@SKYWARN")
- **"Max Replies"** – The number of times your station is allowed to auto-respond for this expect; Enter 0 to temporarily disable the responses, or enter a number, 99=∞

## Edit, Remove, or View an Expect Statement

To edit an Expect statement, highlight the row and right-click. This will load the variables into the fields at the bottom of the window. Make your changes and hit the "Save" or "Cancel" button. To remove an Expect statement, highlight the row and hit the <Delete> key on your keyboard. You will be prompted to verify that you want to remove the row. To view information about an expect statement, including a report on the Callsigns and groups it has been transmitted to, highlight the row and double-click.

## Importing and Exporting Expect Statements

Import and export are simple functions with tab-delimited data. It is recommended that you view the export data to understand the import format. Imported expect statements will overwrite existing statements with the same "text to expect".

## Triggering Expect Responses

For Expect to work, the following needs to happen:

- Multiple operators **should coordinate and practice** to understand this system. There are no default Expect statements. **Conventions may be adopted**, and some are recommended in the tips section below
- You must **DISABLE/UNCHECK** the "Allow sending standard messages without callsign" option in JS8Call settings "General>Behavior" tab, or "E?" requests will not function
- Auto-Respond **must be enabled** in order for JS8Spotter to cause JS8Call to transmit.
- Expect statements are triggered with the "E? " prefix. Note the space. The format is: "E? <Space><Expect Text>"

As an example, say that my station, KF7MIX, is within range (verified with "SNR?"), and has stored the following Expect statement data:

Expect:	Respond:	Allowed:	Max:
CMDS	WX, RIG, ANT, MCF	*	99

To trigger a response to this Expect statement, a station would have to send the following text directed to my station, such as in:

"KF7MIX: E? CMDS"

If station AB1CD sent the above command, it would be received as "AB1CD: KF7MIX E? CMDS", and my station would automatically reply with "KF7MIX: AB1CD WX, RIG, ANT, MCF"

As another example, say that the station AB1CD is within range and has stored the following Expect statement data:

Expect:	Respond:	Allowed:	Max:
WX	75F SUNNY	*,@SKYWARN	99

Imagine also that both AB1CD and myself, KF7MIX, participate in the @SKYWARN group. If I send:

"@SKYWARN E? WX"

Then AB1CD (and any other @SKYWARN participants in range) would receive "KF7MIX: @SKYWARN E? WX" and their station (or multiple stations at once, if others are also setup) would automatically respond with "AB1CD: @SKYWARN 75F SUNNY"

As one final example, imagine three fictional friends: EF2GH, IJ3KL, and MN4OP. They like checking in on one another, and they each store a "STAT" statement with a response just for their friends. Below would be the Expect statement data that EF2GH has stored:

Expect:	Respond:	Allowed:	Max:
STAT	ALL CLEAR	IJ3KL,MN4OP	99

If station IJ3KL sent "IJ3KL: EF2GH E? STAT", then the response would come back as "EF2GH: IJ3KL ALL CLEAR". Likewise, MN4OP could trigger the response. But if AB1CD saw them communicating, and one day tried "AB1CD: IJ3KL E? STAT", there would be no response, as his station is not in the allowed list.

## Sending an Expect Response Manually

You can send Expect responses manually by first right-clicking a row to load it, then clicking the "Send Now" button at the bottom of the Expect window. Fill in the target station or group (which do **not** need to be in the allowed list for manual TX), and optional check box for sending the response as a MSG (which will prompt an ACK from the receiving station.)

Manual Expect responses are primarily intended for transmitting filled out MCFORMS, which will be covered later. However, you may transmit any Expect response manually, at any time.

## Expect Operating Tips

To maintain flexibility and honor varying perspectives on privacy, Expect only does what you tell it to do. It does not expose new universal remotely-activated commands. It will not reveal a list of available commands unless you set that up manually. For this reason, it is valuable to keep some potential practices in mind:

- Expect is primarily intended for groups of operators. These groups should operate off a common set of procedures, such as a published SOP. At the very least, these groups should agree on how to configure their Expect systems so that they know what commands to send to trigger responses.
- It is good practice and highly recommended to setup a "CMDS" Expect statement. This statement should be accessible to "\*" and should respond with a list of Expect commands that you want the general public to be aware of. It could be short or long, and it doesn't have to list all of your commands, just the public ones. Examples of "public" commands might include:
  - WX – store and report your daily weather
  - RIG – store information about your rig here, and use "INFO?" to direct them to "E? CMDS"
  - ANT – store information on your antenna system specifically, instead of cramming it into "INFO?"
  - URL – store a website URL that other operators can visit
  - GRP – store a list of groups you participate in
  - MCF – store a list of any forms you have available for retrieval
- You may include the text "MSG " at the beginning of your responses, if you want to transmit those responses as JS8Call messages. This will prompt an ACK from the receiving station
- You may send E? requests through a relay, but response is delayed for 120seconds to allow for the JS8Call built-in ACK replies and other multiple communications associated with a relay. Simply wait for the station to send the E? response after all the JS8Call chatter is completed

## Pausing the Expect System

You may temporarily pause the Expect system using the "Tools>Pause Expect System" menu item. None of your Expect statements will be lost, but your station will not auto-respond until you un-pause the system. This setting is retained even if you exit the program, but a notification is provided at different points to remind you that Expect has been disabled.

## Expect Blocklist

You may prevent stations from triggering Expect responses by adding them to the blocklist. Access the comma-separated list in the Tools menu.

### 3. Multiple Choice / Micro Forms

JS8Call is a slow mode, tailored specifically to weak signals. Normal forms, as seen in programs such as Winlink and Flmsg, are open format, freeform input intensive, and large (by JS8 standards.) The MCFORMS (which stands for both Micro Forms and Multiple Choice Forms) features of *JS8Spotter* bring forms to JS8Call in a way that is respectful of the mode and of the operators using the mode. MCFORMS aims to:

- Allow operators to craft complex multiple-choice forms that convey a great deal of information
- Limit transmitted characters by modeling forms around choices rather than freeform information
- Provide for additional freeform information in comments and additional transmissions
- Model possibilities through included forms
- Provide a resource for publicly publishing forms, so that transmission meaning is never obscure
- Include an internet gateway for groups who want to submit gathered forms to a central location

#### Creating Form Templates

The plain-text (ASCII) format for MCFORMS templates is designed to be quickly human readable.

<b>MCFORMS Template Format Requirements</b>	
<b>First Line</b>	<p>The first line of the form text file <b>must</b> contain:</p> <p>&lt;Title text&gt;&lt;Pipe&gt;&lt;Form ID&gt; (i.e. "Test Form F!000)</p> <ul style="list-style-type: none"> <li>• The &lt;Title text&gt; may be anything but should be short and to the point</li> <li>• &lt;Pipe&gt; is simply the   character, also called "vertical line" or "vertical bar"</li> <li>• &lt;Form ID&gt; must be the letters "F!" followed by 3+ characters</li> </ul>
<b>Question Lines</b>	<p>Questions are <b>opened</b> by a line in this format:</p> <p>&lt;Questionmark&gt;&lt;Space&gt;&lt;Text&gt; (i.e. "? What is your favorite color?")</p> <ul style="list-style-type: none"> <li>• Start the line with a question mark and space, which won't be displayed</li> <li>• A Question Line must open a question before Answer Lines start</li> <li>• Question Lines need not end with a "?", often, a ":" is more appropriate.</li> </ul>
<b>Answer Lines</b>	<p>Answers are <b>added to an open question</b> in this format:</p> <p>&lt;At Symbol&gt;&lt;Character&gt;&lt;Space&gt;&lt;Text&gt; (i.e. "@1 Red" or "@R Red")</p> <ul style="list-style-type: none"> <li>• The answer &lt;Character&gt; you choose can be a number or letter</li> <li>• You don't have to have any special order to your answer &lt;Character&gt;s</li> <li>• The &lt;Character&gt;s will be transmitted over the air</li> <li>• The &lt;Text&gt; can be anything you choose</li> <li>• Answer Lines are associated with the Question Line that precedes them</li> </ul>



<b>Headers</b>	Header lines must be above question lines, and are in this format: <Up to three exclamation marks for level 1,2,3><Space><Text>
<b>Instructions</b>	Instruction lines must be above question lines and are in this formation: <Period><Space><text>
<b>Comments</b>	Any line beginning with "#" will be ignored as a comment. <b>The first line of the text file must not be a comment!</b>
<b>Please Note</b>	It is recommended that you use a basic text editor (such as notepad++ for Windows) and use only ASCII characters in your form files (no icons, emojis, etc).

Creating multiple choice forms requires flexible thinking. Many forms that would otherwise require the operator to type out a brief response, may be broken down into one or more multiple choice answers that require less dependence on operator training by creating response boundaries. Take this theoretical example:

<p><b>Standard Form Question:</b> 1. Status of utilities at your QTH:</p> <p><b>Freeform Answer TXed:</b> 1. "My power is on, but my gas is shut off. We've got city water, but it's only working part of the day. The internet is down but my cell phone is working some of the time, I can text and that's about it. There are radio and TV stations working too, I think they're all up." (269 characters)</p>	<p><b>MCForms Questions (see form F!301):</b></p> <ol style="list-style-type: none"> <li>1. Operator reporting for:</li> <li>2. Phone landlines functioning?</li> <li>3. Cell phone functioning?</li> <li>4. Local AM/FM/TV stations broadcasting?</li> <li>5. Internet functioning?</li> <li>6. Public water works functioning?</li> <li>7. Commercial power functioning?</li> <li>8. Natural gas supply functioning?</li> <li>9. NOAA Weather Radio functioning?</li> </ol> <p><b>MCForms Answers TXed:</b> F!301 <i>143132111</i> #ABCD (21 characters; to decode the italic responses to questions 1-9, see the image in the next section, or view form F!300 in the software)</p>
---	--

The above example doesn't show the answers available to the operator, but you can see them in form F!301 included with *JS8Spotter*. You can never replace every free form question with multiple choice questions. However, it is possible to convey a great deal of information cleanly and uniformly by breaking questions up.

## Filling Out and Posting Forms

In the *JS8Spotter* main window, using the "Tools>MCForms – Forms" menu item, you may access a list of all the forms that are located in the "forms" subfolder in your *JS8Spotter* installation directory. A series of forms have been included by default. You may add to these by creating new text files for the appropriate format, in this folder.

By selecting a form from the sub-menu of "Tools>MCForms – Forms", you will be presented with a window like this one:

**F!301 -- Field Situation Report**

Operator reporting for:	1 Reporting conditions for self in QTH location
Phone landlines functioning?	4 Unknown
Cell phone functioning?	3 Partially functional, texts only
Local AM/FM/TV stations broadcasting?	1 Yes, fully functional
Internet functioning?	3 No
Public water works functioning?	2 Partially functional
Commercial power functioning?	1 Yes, fully functional
Natural gas supply functioning?	1 Yes, fully functional
NOAA Weather Radio functioning?	1 Yes, fully functional

Form Comment:

To fill out the form, select the best answer to each of the questions presented. Add a comment at the bottom as needed, keeping it as brief as possible.

To post a form, click "Post Form to Expect". If you have previously posted this same Form ID, it will be overwritten (you will be notified of this fact each time you post).

You may load a previously posted form from the Expect subsystem by clicking "Load Posted Expect Form." This will allow you to edit your answers.

If you wish to remove a posted form, simply delete the Expect statement.

## Transmitting a Form Over the Air

Forms are posted to the Expect subsystem, and are automatically transmitted upon request. The format for transmitting forms over the air can be easily viewed after posting a form to Expect, and is:

F!<three characters><space><form response><comment and or space(s)><datecode>

Transmission may be preceded by "MSG" if desired, but it must be added manually. To transmit a form immediately, post the form then visit Expect, load the form into the edit fields, and click the "Send Now" button. The <datecode> will be explained later in this section.

## Form Responses

JS8Spotter will gather all forms that it sees transmitted over the air. These responses are available by using the "Forms>MCForms – Responses" menu item in the main window. You may manually import form responses.

View Form Responses: View All Form Types All Time

From	To	Form #	Form Responses	Message	Timestamp	GW	Received
W5SB	W0RQ	F!103	N115	410000		✓	2023-10-18 20:48
AD00M	W0RQ	F!103	N114	#1537 73. MSG	10/18 10:06	✓	2023-10-18 20:01
KFTMA	W0RQ	F!103	N124	3350	10/18 14:28	✓	2023-10-18 19:36
W0RQ	KFTMA	F!100	8343332120	00 JOB	10/13 17:16	✓	2023-10-13 22:43
W0RQ	KFTMA	F!100	8343332120	00 JOB	10/13 17:16	✓	2023-10-13 22:32
W0RQ	KFTMA	F!103	N153		3/8 11:06	✓	2023-10-13 18:20
K00M00	W0RQ	F!103	N125	TA FRANK	10/4 12:08	✓	2023-10-04 21:53
W0RQ	K00M00	F!103	N155		10/4 13:34	✓	2023-10-04 21:46
W0RQ	KFTMA	F!100	8345452229	ED	9/10 17:00	✓	2023-09-28 00:26
W0RQ	W0RQ	F!100	8343332120	80709273023	9/10 10:36	✓	2023-09-27 23:43
W0RQ	K3RTA	F!100	8343332120	80709273023	9/10 10:36	✓	2023-09-27 23:34
W0RQ	W0RQ	F!100	8343332120	80709273023	9/10 10:36	✓	2023-09-27 22:39
W0RQ	W0RQ	F!100	7345452229	OP ED	9/26 17:14	✓	2023-09-27 22:33
W0RQ	W5SB	F!100	8343332120	80709273023	9/10 10:36	✓	2023-09-27 20:41
W0RQ	K00M00	F!100	8343332120	80709273023	9/10 10:36	✓	2023-09-27 19:41

Import Export All Gateway:  Save GW Forward

In the Responses window, you may:

- Filter by Form ID and/or time received
- View information at a glance, including whether the form was submitted to a Gateway ("GW")
- Manually Import/Add form responses to the list in the format:
  - FROM<space>TO<space>FORM#<space>RESPS<space>MSG<space>TIMESIG
- Use the "Export All" button to save responses in a spreadsheet-type format
- Double-click a response to view/save a formatted human-readable version, select and right-click for GUI
- Highlight one or more responses and hit "Delete" to remove them
- Forward a response to another station or group

## Response Gateway

In this context, the Gateway is a user-provided web-based application which receives specific HTTP/HTTPS requests and processes the included data, storing and/or performing other functions with that data. A very basic sample gateway is available for testing (with source) on [kf7mix.com](http://kf7mix.com).

The Gateway is an optional feature intended for groups of operators who wish to leverage the internet to increase the speed at which the group may communicate forms, or integrate with other systems.

In the Response window you have the option of setting a Gateway URL or leaving the field blank. If you provide a properly formatted URL, forms received by *JS8Spotter* will be submitted to that URL. For example:

<b>Sample/Test Gateway URL:</b> <a href="https://kf7mix.com/mcfgateway_submit.php">https://kf7mix.com/mcfgateway_submit.php</a>
<b>Example Posted Request:</b> <a href="https://kf7mix.com/mcfgateway_submit.php?fromcall=AB1CD&amp;tocall=KF7MIX&amp;typeid=F!301&amp;responses=143132111&amp;msgtxt=HELLO&amp;timesig=#CJIB">https://kf7mix.com/mcfgateway_submit.php?fromcall=AB1CD&amp;tocall=KF7MIX&amp;typeid=F!301&amp;responses=143132111&amp;msgtxt=HELLO&amp;timesig=#CJIB</a>

Groups may choose to keep their Gateway URL private, and include additional key/value pairs in the Gateway setting (by adding "?name=value" to the Gateway URL) to provide basic authentication. For example:

<b>Fictional Gateway URL with added key/value pair:</b> <a href="https://ab1cd.com/spotter_gateway.php?operatorcode=719015221">https://ab1cd.com/spotter_gateway.php?operatorcode=719015221</a>
<b>Example Posted Request:</b> <a href="https://ab1cd.com/spotter_gateway.php?operatorcode=719015221&amp;fromcall=AB1CD&amp;tocall=KF7MIX&amp;typeid=F!301&amp;responses=143132111&amp;msgtxt=HELLO&amp;timesig=#CJIB">https://ab1cd.com/spotter_gateway.php?operatorcode=719015221&amp;fromcall=AB1CD&amp;tocall=KF7MIX&amp;typeid=F!301&amp;responses=143132111&amp;msgtxt=HELLO&amp;timesig=#CJIB</a>

## MCFORMS Datecode

The datecode at the end of an MCFORM transmission provides an approximate low-resolution timestamp in a human-readable format with as few characters as possible over the air. It starts with "#" and the 4 characters following are interpreted as:

First Character	Month, represented as a single letter: A-L = 1-12
Second Character	Day, represented as a single letter or number: A-Z = 1-26, 0-4 = 27-31
Third Character	Hour, represented as a single letter: A-W = 0-23
Fourth Character	Minute, represented as a single letter, 2min resolution: A-Z, 0-3 (A=0, B=2, C=4, etc.)

## 4. Additional Features

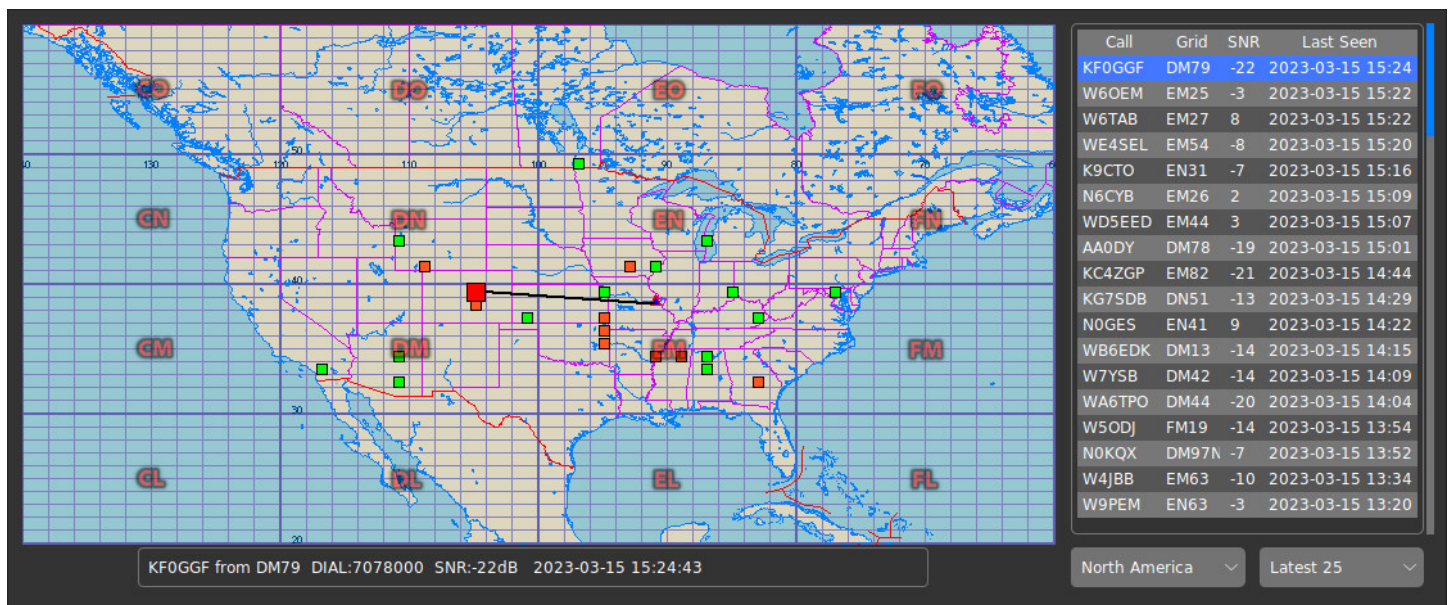
### APRS Commands

JS8Call supports APRS-IS integration natively. APRS commands transmitted in the proper format may be received by JS8Call stations and forwarded through the internet to the APRS system. JS8Call and *JS8Spotter* provide mechanisms for emails, SMS, grid reports, and POTA Gateway.

*JS8Spotter* makes entering APRS commands easier by providing simple fields to fill out. These are found in the "Tools" menu of the main window. You may use these to generate the APRS statements, which may be copied or transmitted directly.

### Simple Offline Maps

JS8Call transmits maidenhead grid coordinates regularly with a variety of transmission types. Because these grids are readily available and already included in other transmissions, it makes sense to provide a way to visualize them. Visualizing stations that you can hear can help you understand current propagation.



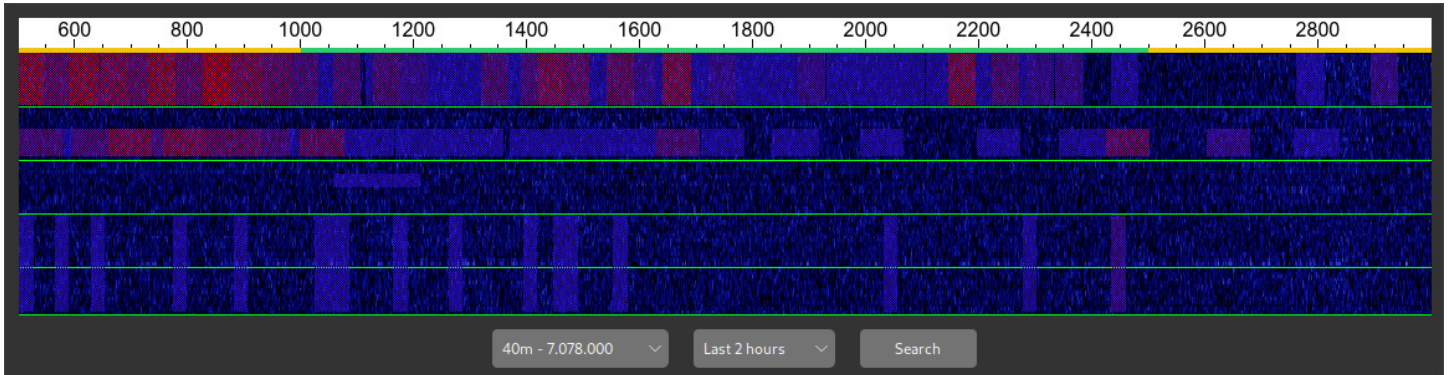
The map feature is located in the "Forms" menu of the main window.

- Maps for North America, Europe, Australia, and Indonesia are included. You may add additional maps in the code, or email to request they be added in future releases.
- You may filter results by age, the latest 100 are viewed by default
- Make sure your own grid location is set in the settings, as described in this manual
- Double-click a line to view details on that line
- Select a line and right-click or hit <Delete> to remove an item
- Only the most recent grid report of any given callsign will be shown
- Locations are very approximate (4-letter grid)



## Visualize Waterfall

For operators who like to leave their station running unattended, it can be difficult to select a portion of the waterfall that isn't busy. This feature will help you visualize JS8Call traffic over time on a simulated static waterfall.



Select the band and time-frame desired.

- Only bands you're actively using in JS8Call will gather data for this visualization
- The most recent signals are red, the oldest signals are blue
- Signals are separated by speed: normal at the top, fast next, turbo, then slow at the bottom
- The heartbeat portions of the band will always be the busiest
- Use "Search" to filter/search by callsign

## CommStat Compatible Features

JS8Spotter includes some basic CommStatOne compatibility features, including:

- The ability to capture, view, forward, and export received CommStat Status Reports (statreps)
- The ability to fill out and transmit a CommStat formatted statrep
- The ability to fill out and transmit a CommStat compatible net checkin

These features are meant to provide basic compatibility only, and are in no way designed to replace the CommStatOne software package.

## Simple Database Search

A very basic database search tool has been included in the Tools menu. This feature will allow you to quickly search through specific database tables based on text you provide. Users who do not wish to install free database software such as SQLiteStudio or DB Browser for SQLite may use this basic search instead.

## Database Trim

Running spotter continuously over long periods of time, especially with large profiles, may cause your database file to become quite large. If you notice a slowdown in JS8Spotter on your system, you may use the database

trim feature to reduce the size of your database file. The trim feature will simply remove all records which are older than 6 months. It is recommended that you make a backup of your database file before trimming.

## Getting Additional Help

Additional help, insights, and training videos are available online:

<https://kf7mix.com/js8spotter.html>