

Serving Amateur Radio for over a decade



Daniele Pistollato (Lele)
IW3HMH



Terry Genes
G4POP

INTRODUCTION

JT (Joe Taylor) based applications use a special UDP message protocol to share information with listeners. One of those messages is the currently decoded calls, another is the logged contact when you save a QSO in the JT program.

There are many others messages containing various information (Data) being shared by the various different software applications.

Once set up to read (Receive) messages via UDP, Log4OM searches for the 'QSO' message to be logged.

IMPORTANT NOTE:

When an application is sending a UDP message (not a broadcast/multicast message) as JTDX or WSJT-X, that message is 'Captured' by the first application that reads the UDP data from the exposed network, no other application will now see that message, its gone!

Once the message has been captured by the first application it is no longer available for other software to read, similar to putting a ten dollars bill on floor in a supermarket. The first person notices and collects the US\$10 leaving nothing for any other shopper in the supermarket to find!

Log4OM, Gridtracker, JAlert and other applications that listen for a UDP message from JAlert are competing to receive the message first. That means that sometimes JAlert can get the message first, sometime GT, sometime Log4OM. This is not good behaviour because if Log4om 'Captures' a message that is of interest to JAlert, then JAlert will not receive the info. In the same way, if GRIDTRACKER 'Captures' a QSO message, Log4Om will receive the message and cannot log it.

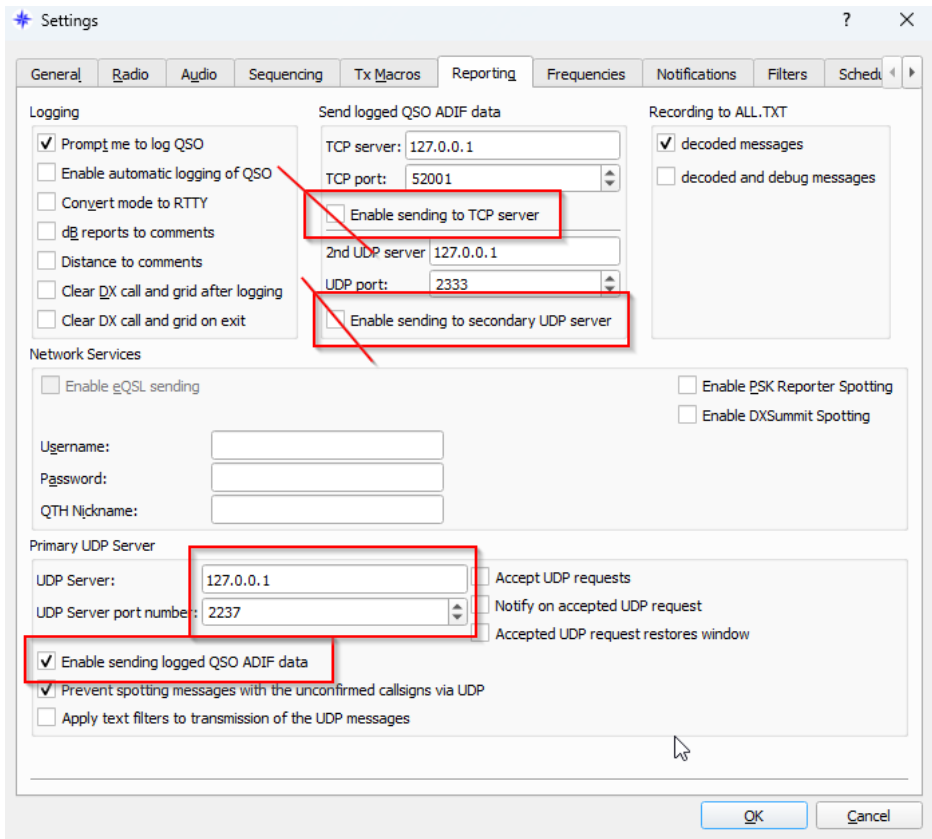
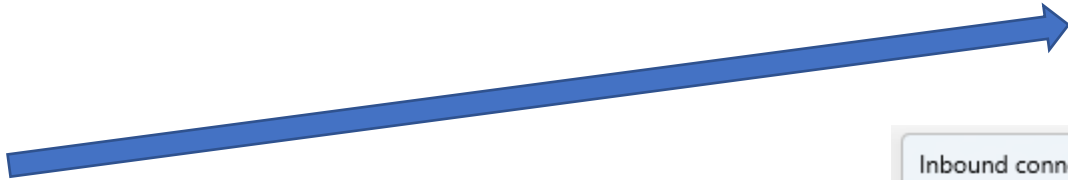
That's why, when configuring multiple JT applications, ONLY ONE APPLICATION SHOULD BE CONFIGURED TO USE JT UDP MESSAGES, and this application should take care of REBROADCASTING the relevant information to other applications that are waiting for the information.

Configuration 1: Log4OM integrated with one JT application (WSJT/MSHV/JTDX)

CASE 1: JT MESSAGE VIA UDP

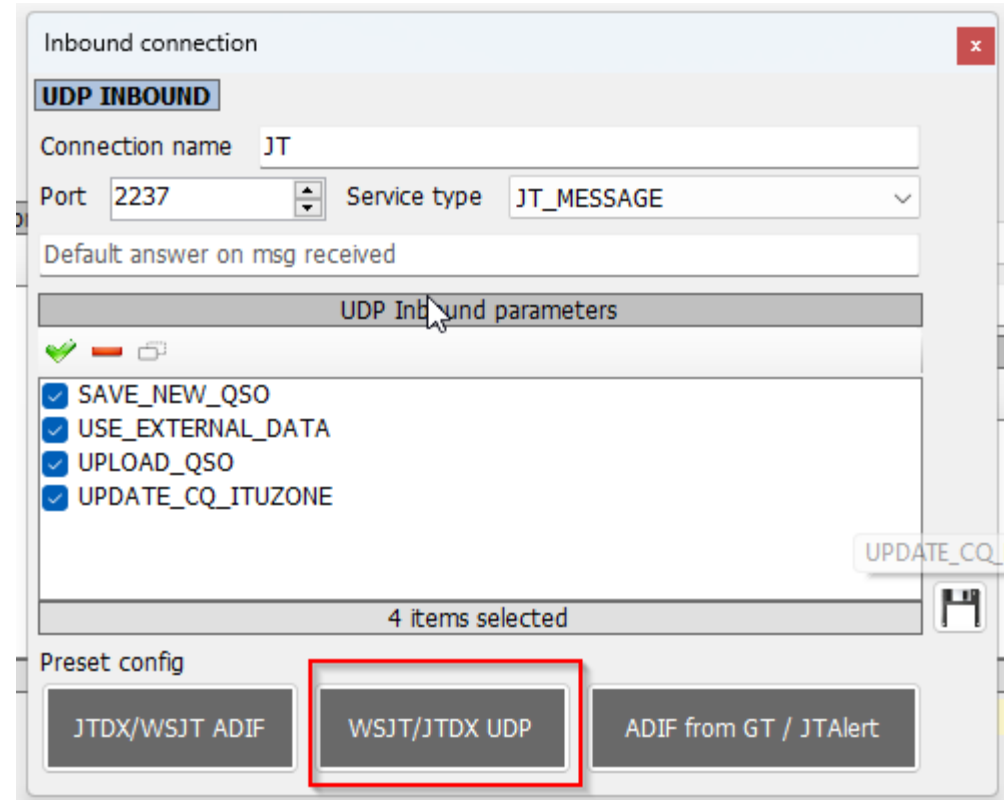


JT APPLICATION



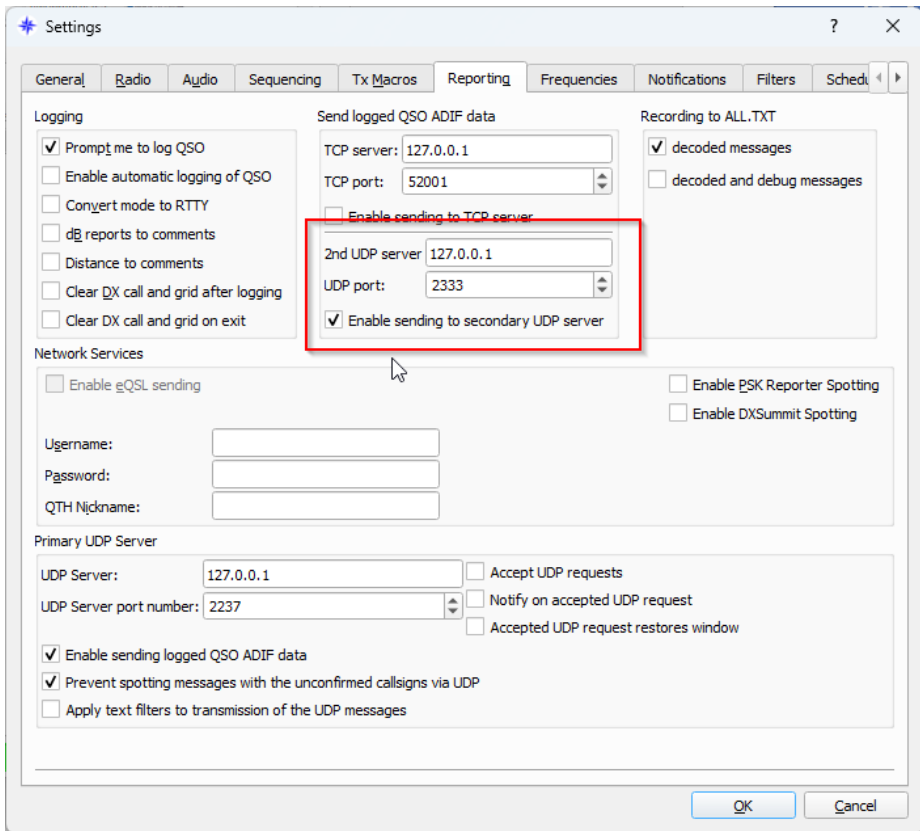
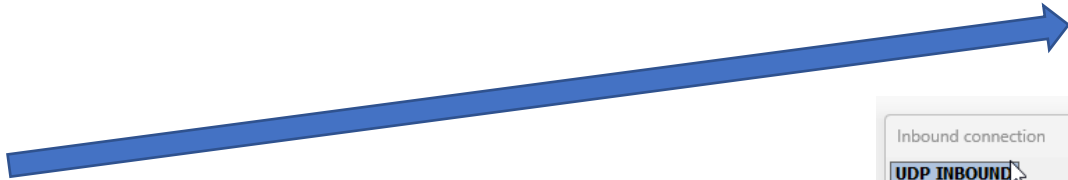
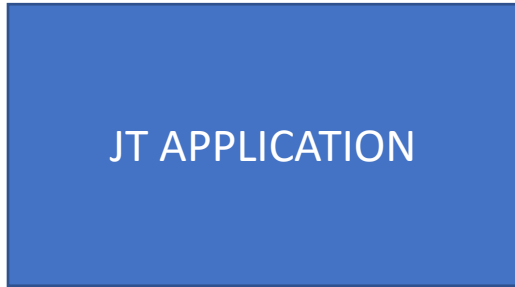
In this scenario Log4OM is the only UDP receiver, Log4OM receives the QSO directly using the JT UDP PROTOCOL that contains logged QSO data.

Users should configure a direct WSJT/MSHV/JTDX UDP connection as shown, Log4OM will then receive the QSO's



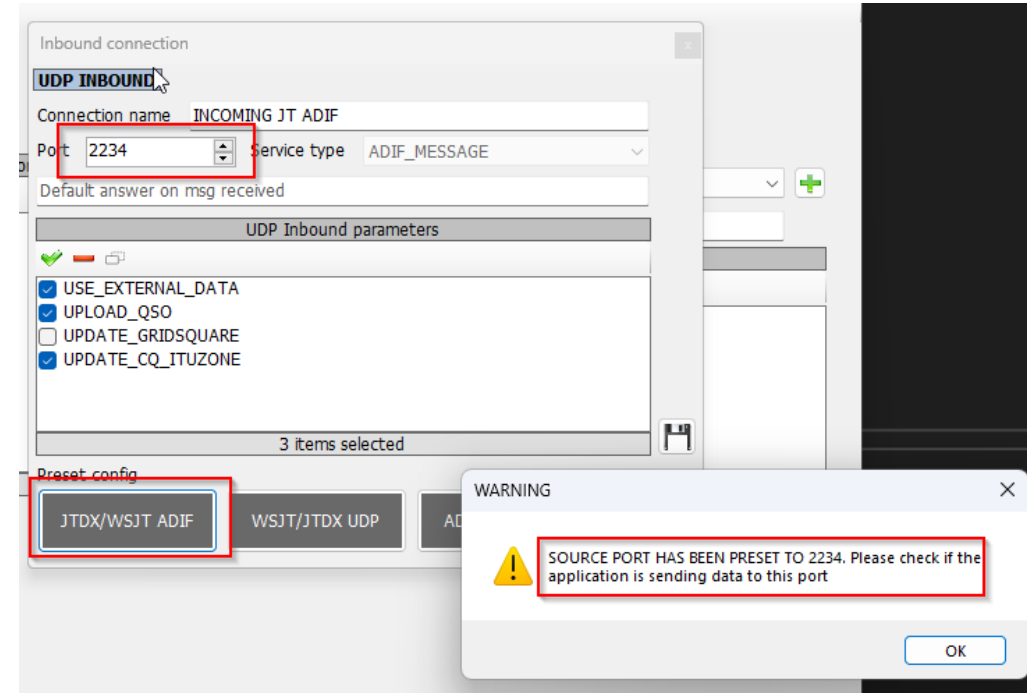
Configuration 1: Log4OM integrated with one JT application (WSJT/MSHV/JTDX)

CASE 1: ADIF MESSAGE



Although not recommended in this configuration, the user may want to use a specific ADIF message to log QSO's into Log4OM. In that case configure an ADIF_INBOUND message into Log4OM listening on port 2333

The Log4OM default port should also be updated to 2333 instead of 2234



PLEASE NOTE:

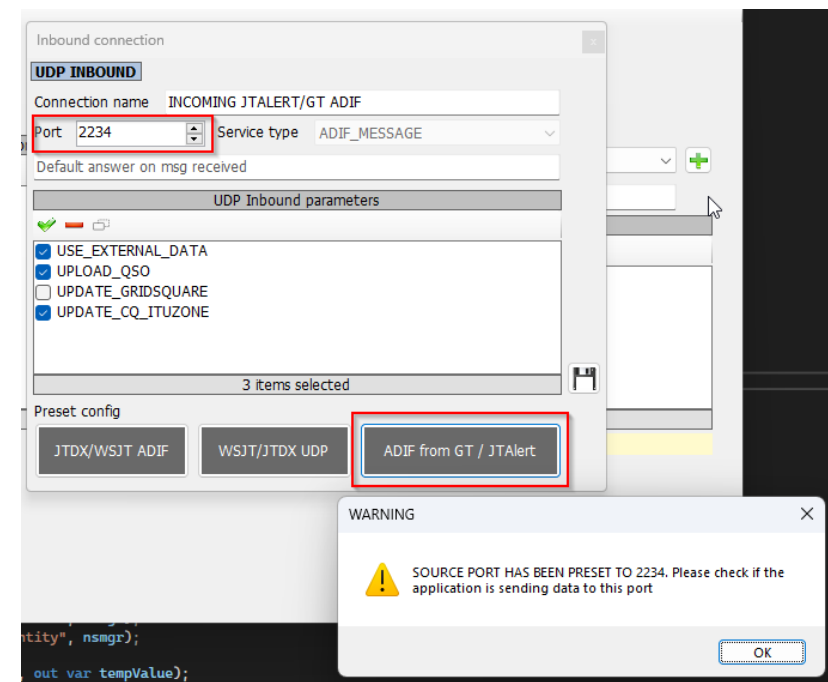
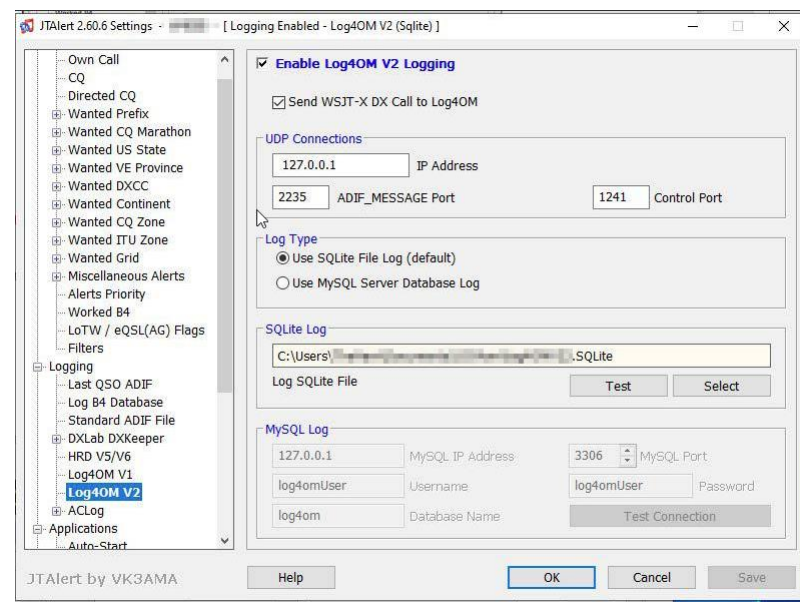
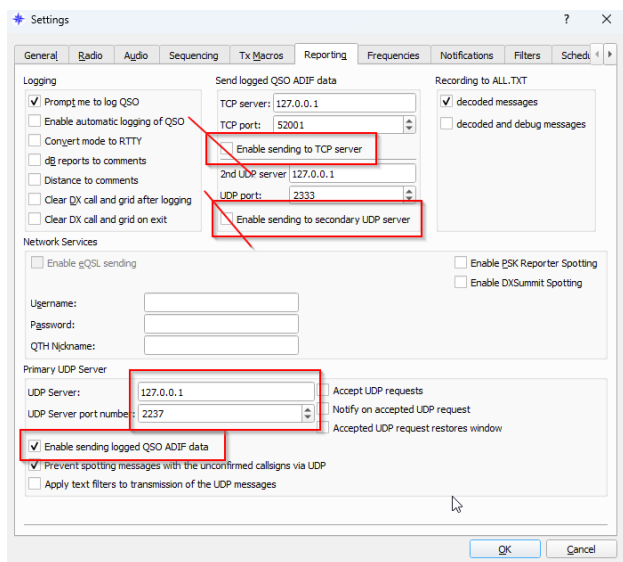
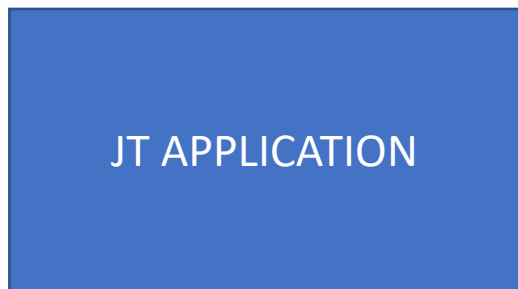
DO NOT ALSO CREATE A JT CONNECTION. If you create also a JT CONNECTION

The QSO will be received twice by Log4OM, potentially causing overhead (The QSO will not be logged due to duplicate keys check, but there will be a lot of errors in the program log)

Configuration 2: Log4OM with JT software and another application (GRIDTRACKER or JTALERT)

In this situation, only one application should get the JT message, use the information and then relay the relevant data to other applications.

The simplest way is to have JTAAlert or GridTracker, to receive JT UDP messages. JTAAlert should then forward/relay the QSO data to Log4OM via ADIF MESSAGE

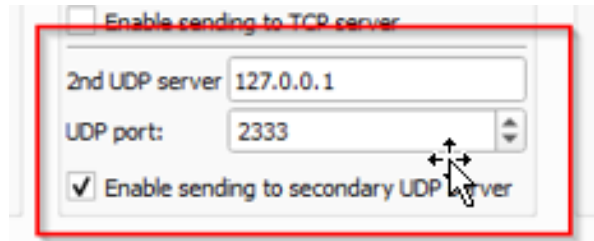


More options:

Once familiar with how UDP messages work it is possible to set up different configurations.

Example:

Enable a secondary UDP ADIF broadcast on JTDX, then put Log4OM reading the message using an ADIF INBOUND CONNECTION on port 2333 while you have JTAAlert reading the JT PROTOCOL messages on port 2237



Enable sending to TCP server

2nd UDP server: 127.0.0.1

UDP port: 2333

Enable sending to secondary UDP server



Primary UDP Server

UDP Server: 127.0.0.1 Accept UDP requests

UDP Server port number: 2237 Notify on accepted UDP request

Accepted UDP request restores window

In this case, JTAAlert should NOT send ADIF messages to Log4OM, because Log4OM will receive them twice (one from JTDX application and one from JTAAlert ADIF message).

If Log4OM is configured to listen on port 2333 for ADIF messages from JTDX and JTAAlert is sending ADIF messages to port 2341 then no program will received or act on the data sent.



UDP Connections

127.0.0.1 IP Address

2235 ADIF_MESSAGE Port

1241 Control Port

BUT...

Other options can be explored, that may be extremely powerful but also create problems...

Both Log4OM and JTAlert are capable of 'listen and discard' a UDP message on port X and 'retransmit' the message to another listener (but only one, or the user will create another competition for the data on secondary broadcast) on a DIFFERENT PORT, Log4Om does this with the 'proxy connection' type.

Basically, it reads the UDP message on port X, uses it and resend the message to port Y, where another application is waiting the «JTDX/WSJT message».

This way, it is possible to create chains between applications, where every application is receiving a JT MESSAGE, uses it and then forward the message on another port to another listener.

So, if you configure:

JTDX to send messages on port 2237



Primary UDP Server

UDP Server: 127.0.0.1 Accept UDP requests

UDP Server port number: 2237 Notify on accepted UDP request

Accepted UDP request restores window

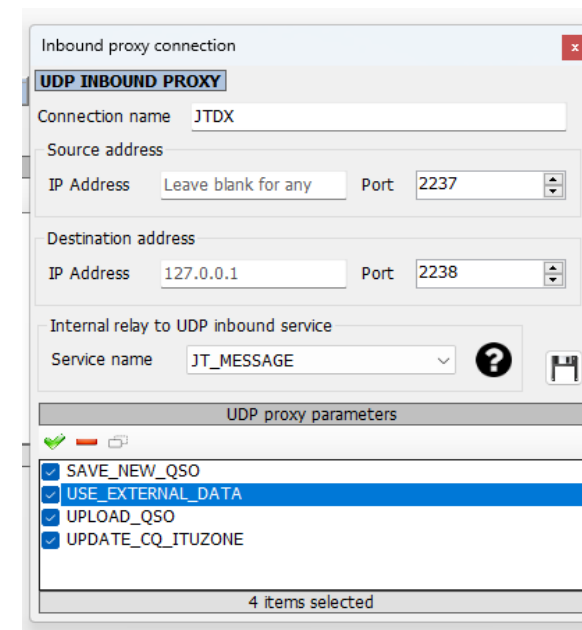
Log4OM to read messages from port 2237, and rebroadcast using proxy feature

To port 2238, JT Alert can be configured to «listen» JTDX on port 2238 instead, and

have a chain of applications working 'one to one' with JTDX

as if they were the only ones. Obviously configuring a JT rebroadcast on

JTAlert to another port you can "connect" GRIDTRACKER or another application.



Inbound proxy connection

UDP INBOUND PROXY

Connection name JTDX

Source address

IP Address Leave blank for any Port 2237

Destination address

IP Address 127.0.0.1 Port 2238

Internal relay to UDP inbound service

Service name JT_MESSAGE

UDP proxy parameters

- SAVE_NEW_QSO
- USE_EXTERNAL_DATA
- UPLOAD_QSO
- UPDATE_CQ_ITUZONE

4 items selected

Last words.....

- These notes are correct at the time of writing (30/05/2023) however changes to external software like JT programs is outside the control of Log4OM and users should seek advice regarding such changes from the other software vendors, not from Log4OM.
- Log4OM does not take any responsibility for damage to a users PC or software while using Log4OM or implementing connections described herein.
- For further assistance use the Log4OM forum located here <http://forum.log4om.com/>

73 & GOOD DX FROM THE LOG4OM TEAM.....