OH1KH's additions to cqrlog 2.1.0 rev 11(!!for beta binary!!)

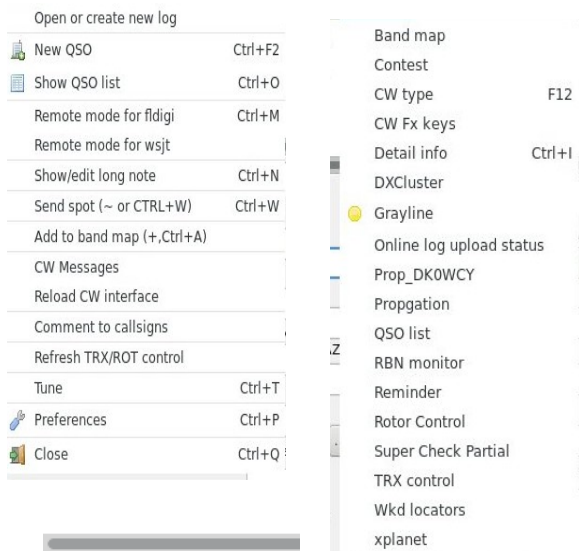# Contents

Selections

I have made some additions to OK2CQR's logging program Cqrlog. Here is a brief description of them.

You can find source code from https://github.com/OH1KH
from source code you will find README.OH1KH that explains what is done and when and also what bugs are found/fixed.
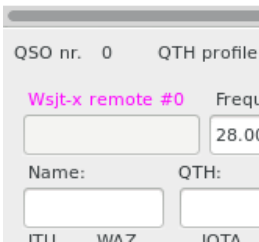

Remote mode for WSJT-X-communication is made via UDP dtatgrams that and is supported from WSJT-X 1.5.0 program upwards.
Cqrlog supports WSJT-X UDP remote mode since version 1.9.1

From NeWQSO/Window selection list you can find now some new properties.

"Wsjtx monitor " is visible only if WSJT-X remote mode is selected. Others can be found from there all the time.

When remote mode is selected it shows last received WSJT-X packet type number after text "Wsjt-x remote" text. Also color of text changes while receiving packets. These will show you that UDP link is alive between these programs.

Fldigi XmlRpc remote

XmlRpc remote connection to fldigi is an alternative way to transfer qso data to cqrlog.

With XmlRpc cqrlog acts as client what is opposite way compared to wsjt-x remote where cqrlog acts like server. Because of this cqrlog can not know if "save to log" button is pressed at fldigi because it tells it's information only when asked.

This leads to situation where fldigi logging can not be used to act like indicator to cqrlog to save qso and **saving must be done by operator on cqrlogs side**.

There are two ways to save qso:

- Press "save qso" on cqrlog's NewQSO form **and** press save on fldigi. You get entry to both logs.
- Press "Save this qso" on cqrlog XmlRpc form. You get qso **saved to cqrlog** and fldigi entry is **cleared**, but **not saved**.

When using XmlRPC:

- You get qrz/HamQth data immediately after adding call to fldigi's call field
- You see qso before/dxcc statistics from cqrlog before firing TX the first time
- You can run fldigi on other computer than cqrlog if they have common network. WiFi, wired, or even internet(when you should use openVPN to make secured tunnel between sites)

Settings
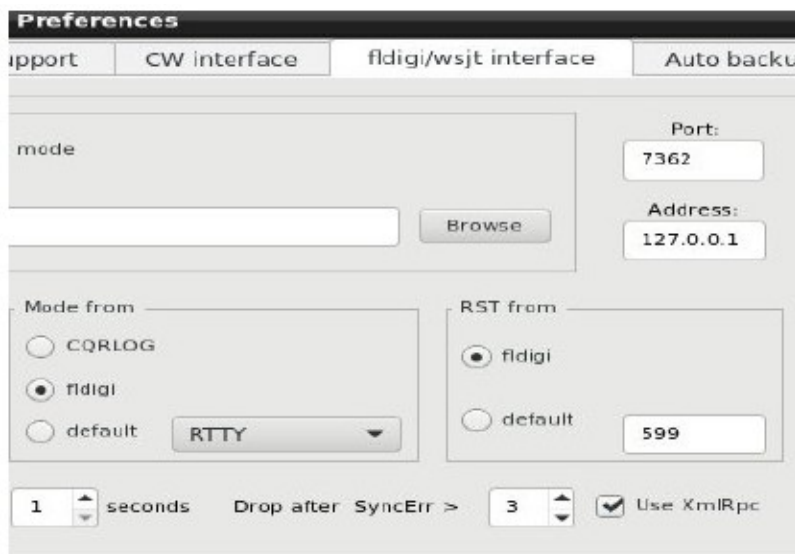
In cqrlog's preferences/fldigi/wsjt interface there are some new settings.
Port and Address for fldigi network connection and selection to use XmlRpC.
With selection you can choose "the old way" for fldigi remote, or the XmlRpc way.
Port number should be ok, but IP address may change depending your needs.
Address is normally 127.0.0.1 (localhost) if you run fldigi on same PC than cqrlog.
If you have fldigi running on other PC put here the IP address of fldigi-PC. There is no IP address validation, so you have to type correctly formatted address.

Parameter "Drop after SyncErr >" automates remote mode closing when fldigi is closed. But as it may affect also (in slow PCs) fldigi started by cqrlog it can be changed. 0 disables, 1-20 means timer rounds until drop.

When closing happens it leaves text "Socket error, check fldigi!" to NewQSO field "Comment QSO" .

If you use fldigi on other PC remember to open firewall at fldigi-PC side for TCP traffic to port 7362 from your local network.

In all cases fldigi must be started from command line with parameter:
fldigi --xmlrpc-server-port 7362

When closing fldgi properly and answering YES to save setup changes the next start should open xmlrpc server without parameters from usual icon.

Use

When you have activated fldigi remote and checked "Use XmlRpc" at preferences,
texts that you type to fldigi logging sheet (or select from monitor with right click) are transferred to cqrlog by polling fldigi with speed you have set on preferences "Try to load QSO from fldigi every X seconds".

When remote for fldigi is opened a new status form is shown. It shows data that have been received from fldigi. If you can't see any changes you have to check port and address settings (both fldigi and cqrlog) and firewall settings.

There is more information about XmlRpc at:
http://www.w1hkj.com/FldigiHelp-3.21/html/xmlrpc_control_page.html

If everything is OK you should see something like this:
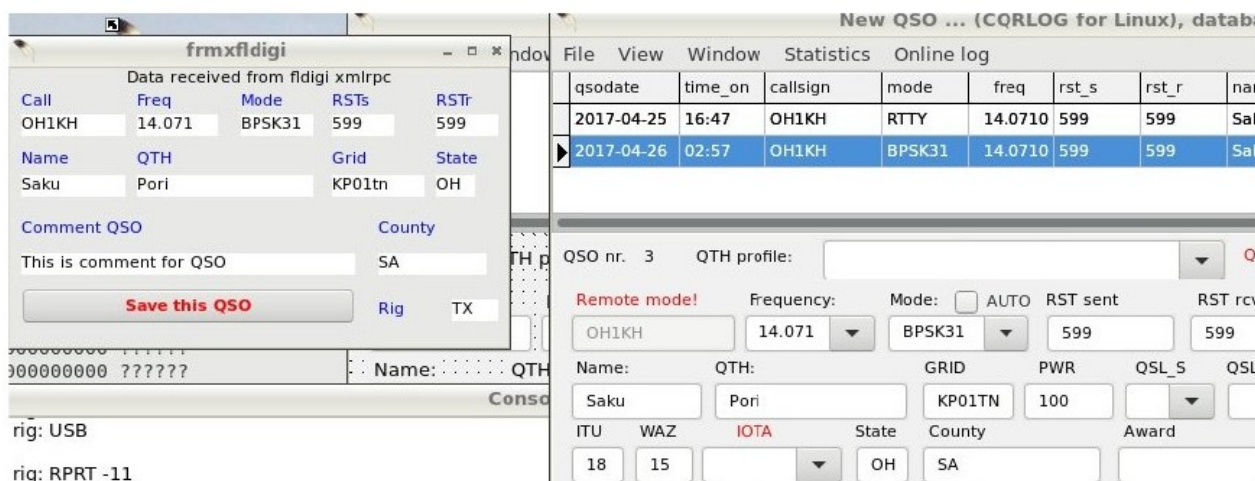
Some notes

- fldigi's **pr** (Province) is transferred to cqrlog's **County**
- fldigi's Country is not transferred
- **Save this qso** is **shown after** you have **call, time on, rst_in, rst_out** and TX fired once it hides again when qso data is cleared.
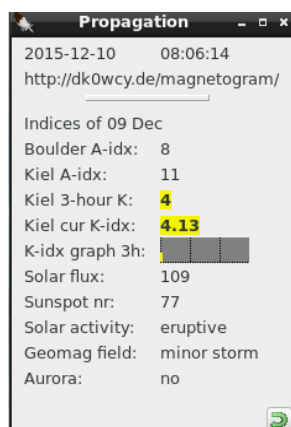- **Save this qso saves only to cqrlog**, fldigi's qso data is just cleared.

Starting fldigi and wsjt-x via cqrlog

You can set preferences so that cqrlog will start fldigi and/or wsjt-x programs for you when you enter remote mode. This is very handy, but doing that has also one risk.

When fldigi or wsjt-x has been started as a child process of cqrlog they will die at the moment cqrlog crashes for some reason.

If you start them manually as processes of their own and then activate remote mode from cqrlog they keep on running if cqrlog dies. You can finish your ongoing qso, start cqrlog again, set remote mode on and then log the qso in normal manner.
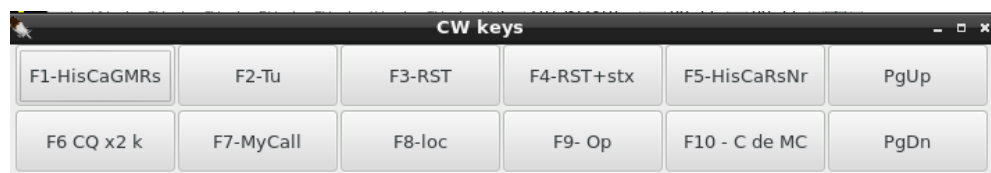
Prop_DK0WCY

This propagation form is an alternative showing details from dk0wcy page with 3 hour graphical display of K-index.

As Boulder information at dk0wcy is one days old top of form shows time when information is read from dk0wcy web page.

"Indices of"- shows the age of Boulder data. Kiel information is up to date as they are produced by dk0wcy.

There is also link to dk0wcy web page. By clicking it page opens to default browser.

CW Keys

CW keys from has new buttons PgUp and PgDn that are not configurable.

They alter CW speed in same way as keyboard keys PgUp and PgDn.

I've done this because of two reasons:

My small HP Compaq mini laptop did not have PgUp and PgDn keys at all on it's keyboard!

When using external keyboard and CW keys-form with mouse you always had to first focus NewQso-form to be active and then press PgUp / PgDn keys. Now you can do it from same focused form that you launch memories with mouse.

Reminder

Reminder is a pop-up form that is **off** by default at program start.

You may initiate reminding based on UTC time or based on passed minutes , but not both.

Closing form will initiate the timer. If your linux has several desktops this pop-up will show up on the top of current desktop. No matter on what desktop your cqrlog is open.
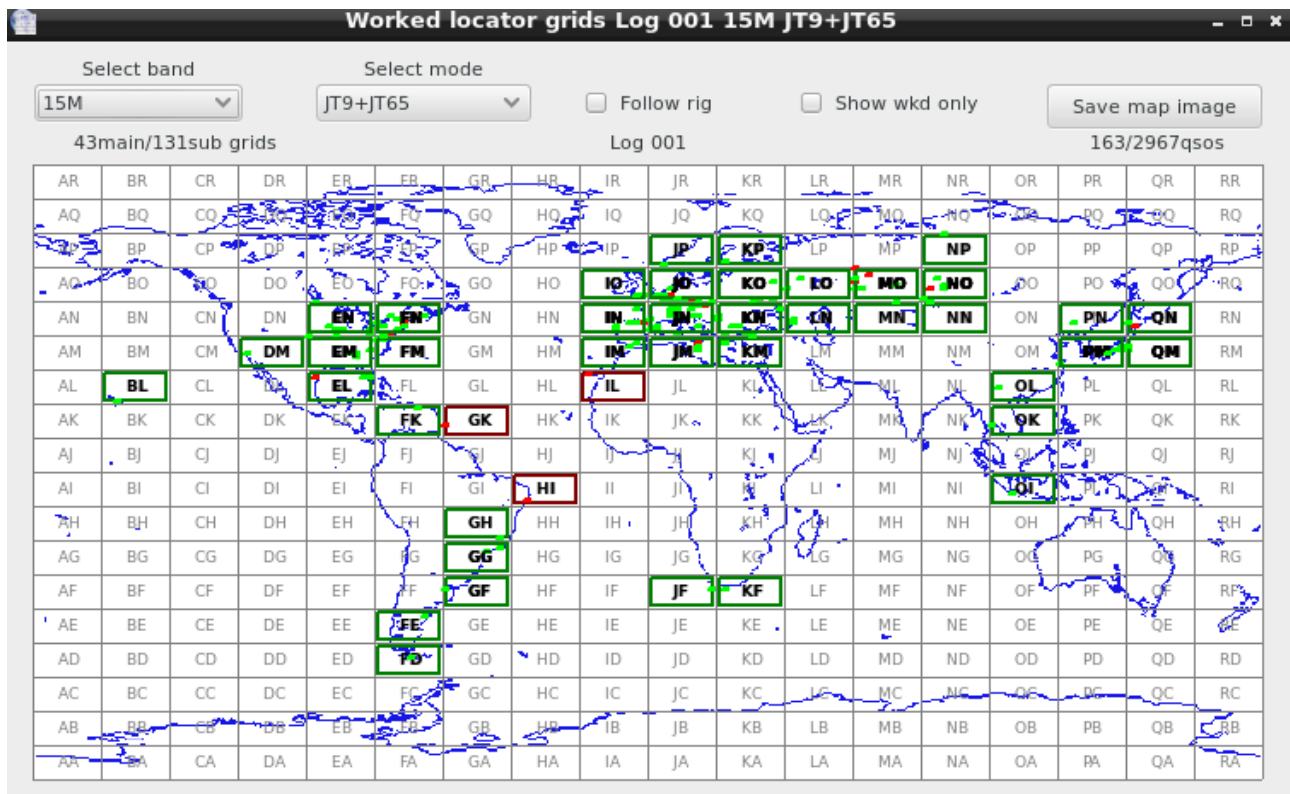
At least it works so with Fedora's LXDE desktops that I have here in use.

You can also write something useful to "Remember to:" memo. (255 characters, one line)

For example "Sked on 20m with OH1XYZ" or " Call wife" or "Pick kids from school" to remember what to do at right time while you are busy with your qsos.

All settings are saved over program restart.

Worked grids

I moved this from my previous project from SourceForge. It was made to show up worked locator grids from JT65-HB9HQX early version's logs. It fits cqrlog nicely and gives graphical view of locators worked in addition of cqrlog's own text based information.

As for JT-mode workers this map us useful also for V-,U-,SHF workers who are collecting locator grids.

Map shows worked grids on selected band and mode, or all bands and modes. Or you can select "Follow rig" and then map follows band and mode that rig is sending to cqrlog via rigctld.

Confirmed main grids show up as green, unconfirmed as red. Subgrids show up with dots inside main grid with coresponding colors.

You can click any main grid to zoom it. Again colors tell you about confirmation.
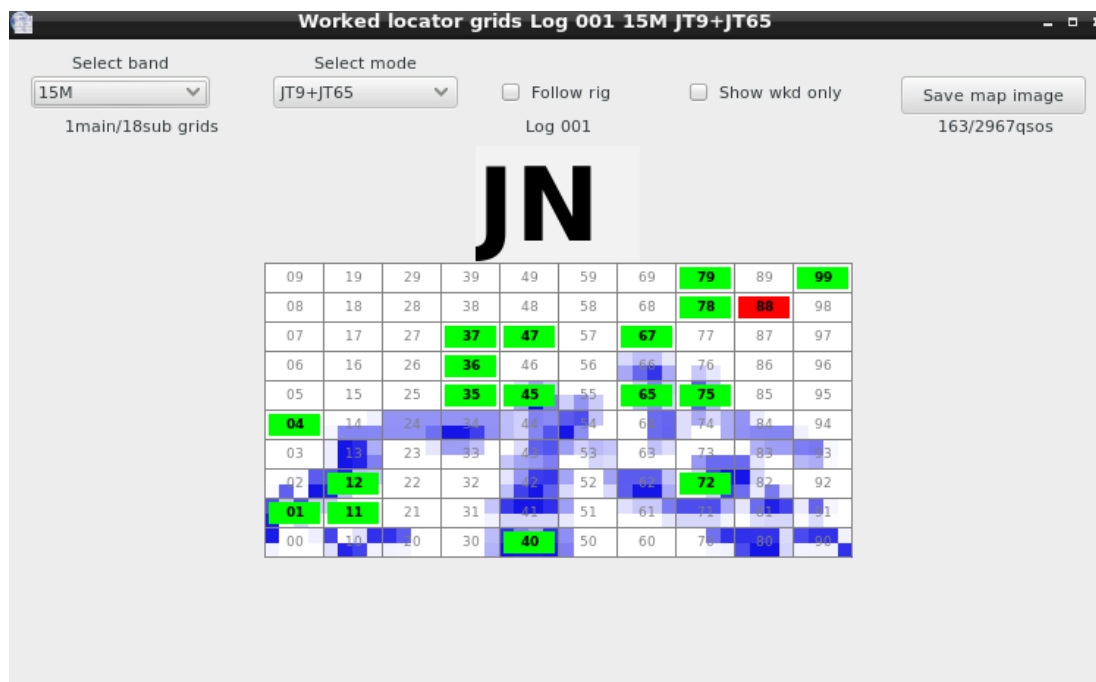
Clicking again on zoomed main grid brings back whole map.
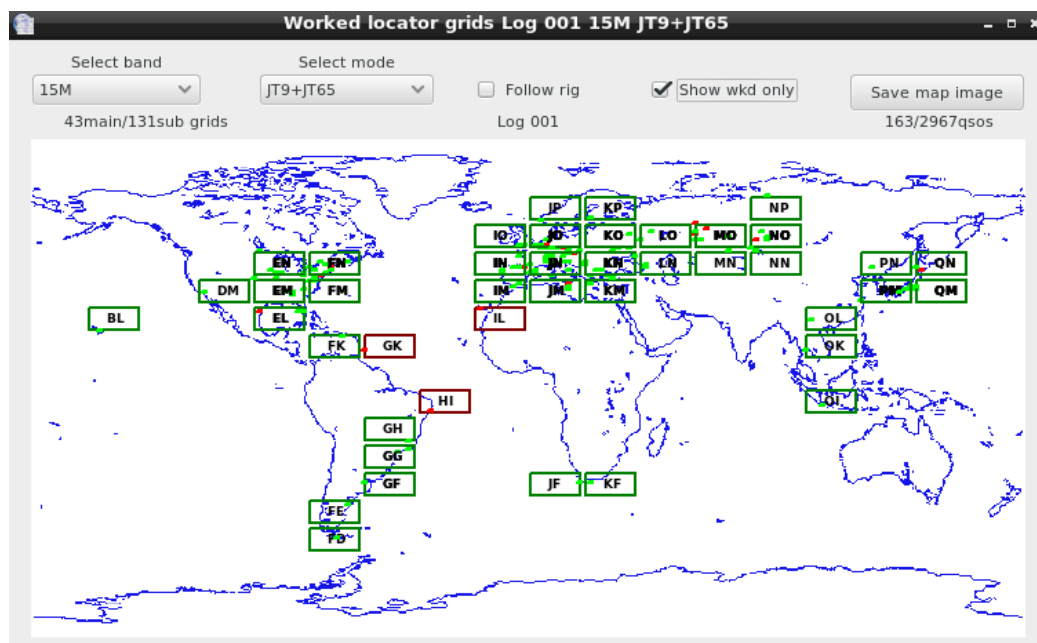
By selecting "Show worked only" removes grids that are not worked from map.

At any time you are able to save currently shown map as an image for other purposes.

Grid map, unfortunately, does not scale. So size is what it is and you have to accept that. Sorry, but it simplified a lot the production of graphics.

Wsjt-x CQ-monitor
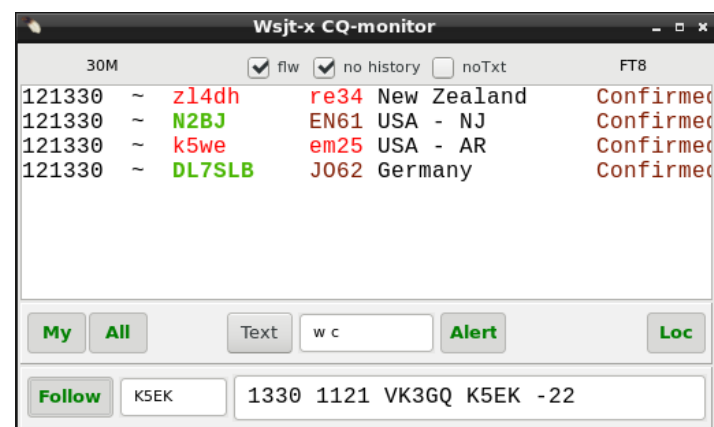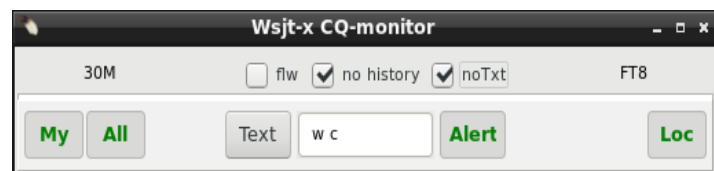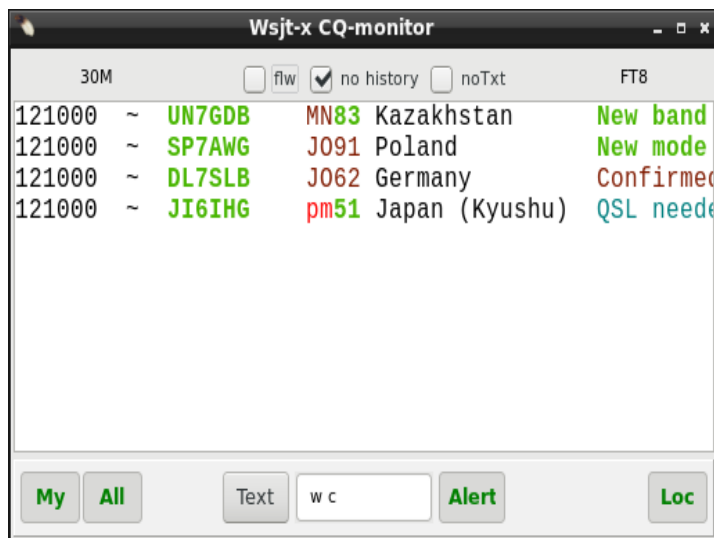
Wsjt-x CQ-monitor form opens when you select "Remote mode for wsjtx" from "File" of NewQSO window.

You can close this window and reopen it later via NewQSO/Window/Wsjt-x monitor that is visible only when remote mode is selected. While it is closed it will not update it's contents.

Only remote logging and call qrz/hamqth seek are used then.

Top line will show band and mode that is in use, Checking **"no history"** will clean monitor on every decoded RX-period. If not checked form will show also older Cqs (scrolling) to see what you missed while having a coffee break :). Only 15 lines, how ever.

If you do not want any information texts, just alerts, you can check **"noTxt"** to prevent richmemo text updates and minimize window (or drop it completely down with (_) while alerts are still working.

Checking **"flw"** new part of monitor opens. You are able to type a callsing, part of it (text-in-text search) or you can push right mouse button down and paint any call from wsjtx main screen or from CQ-monitor and drag and drop callsing to the left side edit box After pressing "Follow" (turns green) the call is checked to be upcase letters and spaces are trimmed from start and end. Next decoding period will show a line where that call is as originating (2nd) callsign and line is **not** CQ or for your call at right side box.

Line contains: **decoding_period_time | delta_frequency | the_message** No alerts are connected to this line. This will make easier to follow a DX who does not stay same frequency and jumps around answering to callers.

Main CQ-monitor line contains:

- time
- mode symbol. Same that wsjt-x uses.
- callsign with color and low/Upcase coding.
- locator grid with color and low/Upcase coding
- country name (cut to 15chrs) . Overlayed with "CQ:xx" and **different color** if station is calling directed CQ like: CQ DX, AS, AF, OC, NA, SA .. or CQ CALLSIGN DX.
  This is set as warning for you to check that you are in directed area before answering to his/her CQ.
  I.E. **In case of *CQ DX* you should be in DIFFERENT CONTINENT as the CQ caller** to be a valid DX for him. Program does not warn you if it thinks you are real DX for caller. Otherwise it does.
- information of DXCC status compared to your logged qsos.

Colors & fonts
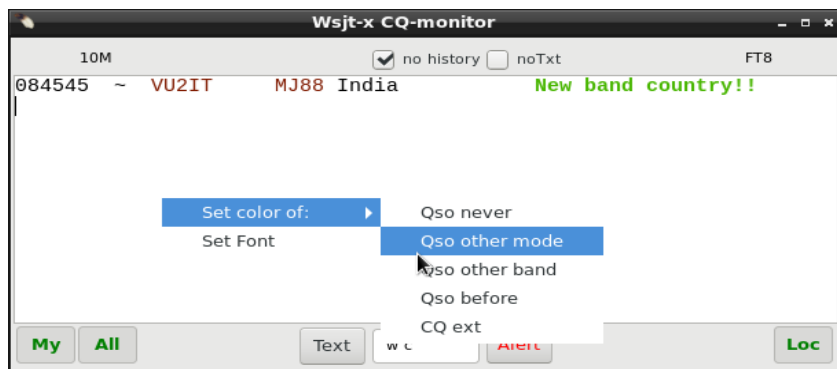
From devel version 2.1.0-109 upwards CQ-monitor has a new design. Window is adjustable and font + size can be changed. Font style can not be changed as now "Qso never" (default Green) is printed also with bold while all other texts are in normal style.

Colors, that user can now change, will tell if station is worked before on **this band and mode**, this band **but not this mode,** some **other band/mode** and **never** on any band/mode.

Same color coding is applied also for locator grid.
Full locator (means 4 first characters of locator) worked before on **this band and mode**, this band **but not this mode,** some **other band/mode** and **never** on any band/mode.

If no hits found for full locator then main locator (2 first letters) are checked same way next . Worked before on **this band and mode**, this band **but not this mode,** some **other band/mode** and **never** on any band/mode.

Calls and locators that are worked before on **this band and mode** are also printed in **lowercase** letters while all others are printed with **UPPERCASE**.
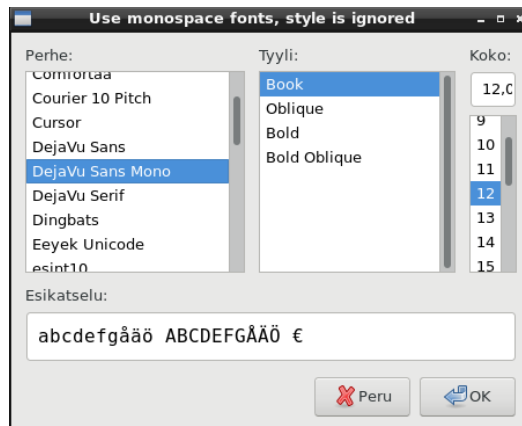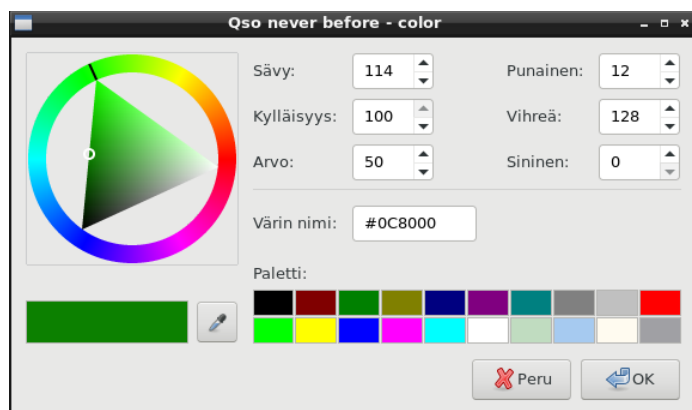
NOTE: Using monospace fonts will keep monitor columns in order.

Behind the scene the SQL routines for finding worked before status are completely rewritten since 2.10(001) version.

Alerts

At bottom there are 3 alerts to select. The priority of alerts is same as numbering here. Smaller number overrides previous having highest priority. Alert is executed only once per every decoding section.

1. "**My Alert**" will alert if someone answers your cq-call. It is useful if you are reading your mails, Facebook or what ever on another desktop while calling CQ with Wsjt-x :)

2. "**All**" will extend "MyAlert" so that all lines that start with your callsign give alerts. There are stations that answer to CQ directly with report, so this wakes you up also then. When using FT8 this might be little annoying but is easy to check off.

3. "**Text [   ]Alert**" will alert if text-in-text if typed text, case sensitive, compared to complete monitor line. **Example:** alert from new country by setting "w c" string to text alert. It fits in to the middle of "New country" string. Text can have spaces, but not at start or at end.
   Press button "Text" to change to call alerts.
   "**Call [   ]Alert**" will alert if text-in-text if typed text, upcase, fits callsign of monitor line. As option, one asterisk can be used at start or end to indicate starts-with or ends-with search. Several search strings can be inserted with comma separation.
   Press button "Call" to change to text alerts.

   Text you have typed is read only after you leave edit box. Then spaces from start and end are removed and in case of "call alert" text is changed uppercase.
   To activate alert press **"Alert"** to turn it's text green. (it also takes you away from edit box)

4. "**Loc Alert**" will alert when there is new main grid that you have not worked.

To make alerts do something you have to add a script file ~/.config/cqrlog/voice_keyer/voice_alert.sh
This script gets parameter $1 of "my","loc","text" or "call" depending of alert given.
By using these parameters with your script you can play sound that you want, or do anything else you like to happen when alert is issued.

NOTE: Try to keep script running time as short as possible.
voice_alert.sh:

```
---------------------------------------------------------------------------
#!/bin/bash
# //audio file name (prefix) played on alert
# //can be:'my'  = ansver to my cq,
# //      'loc' = new main grid,
# //      'text'= text found from monitor line
# //      'call'= text fits to the callsign
# // create files you want to be played
# scirpt is seeking names with '.wav' suffix! Change if needed
#select audio card(if needed) and play alert message
aplay ~/.config/cqrlog/voice_keyer/$1.wav
```

Starting qso from wsjtx-monitor

You can start Wsjt-x qso by double click a CQ-monitor's line. Wsjt-x will move to callers QRG and initiate TX.
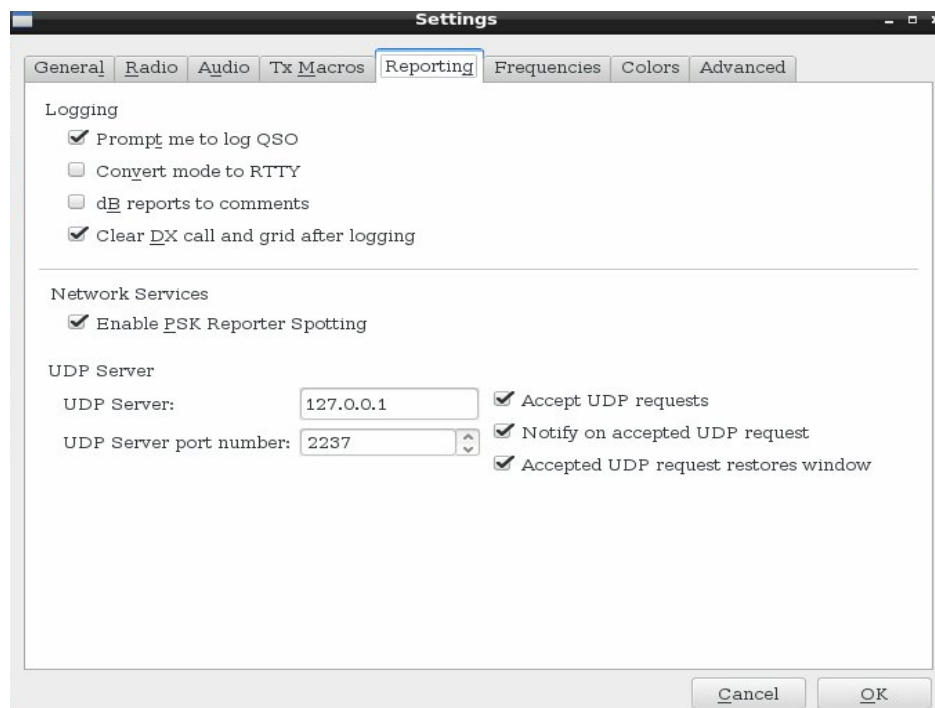
This requires Wsjt-x's **Configuration/Settings/Reporting** to have at least "**Accept UDP requests**" selected.

Setting also " Accepted UDP request restores window" will return focus from CQ-monitor back to wsjt-x main window.

A line when someone has answered to you (your call is first at Wsjt-x data line) has "=" sign just at start of callsign and locator is marked **<!!>.** Clicking that line again does not continue qso.

Wsjt-x remote does not allow this kind of command preventing automated qsos. So at this case you have to go to Wsjt-x screen and initiate report sending by yourself. Or set "auto seg" at wsjt-x/FT8.

Either qso is started from CQ-monitor, or Wsjt-x screen itself, the opponent callsign is added right away to New QSO's callsign field. This way you can see right away if you have had qso with this station on other modes/bands and also all other information like DXCC status, QRZ/HamQTH information etc.

Logging qso to cqrlog

When you either press "Log QSO" at wsjt-x main window, or get it open automatic when sending 73 by checking Wsjt-x's **Configuration/Settings/Reporting/Prompt me to log qso,** you will have wsjt-x's logging form open.

Check it's information, add power and remarks if needed.

After finishing with it press it's OK-button. Then information is transferred to cqrlog.

NOTE: Do **NOT press** cqrlog/NewQSO/Save QSO [enter] - button to save qso!

If you have enabled auto search from QRZ.com/HamQth.com in cqrlog's preferences all information is fetched during your first transmit (report sending period).
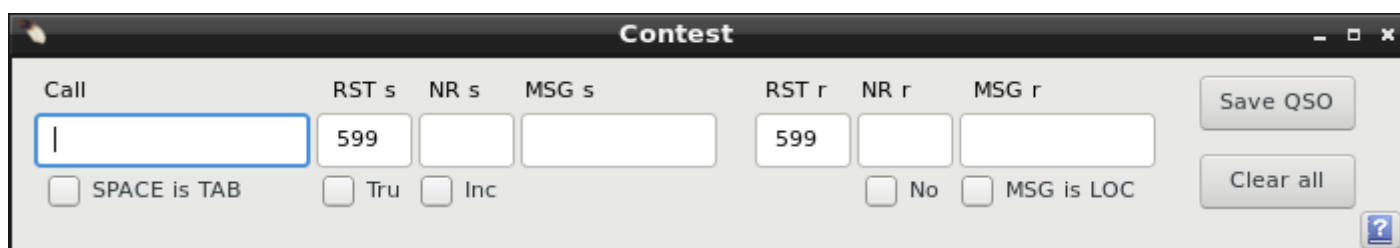You may alter that information in NewQSO form and it is saved along with wsjt-x logging information.

In case that fetched data has same, but longer locator than wsjt-x qso data has the longer is logged to cqrlog.
I.E wsjt-x data gives KP01, but QRZ.com KP01TN, the longer (more complete) is logged.
If locators differ wsjt-x data is used as it might be a portable or other special qth.

Contest support

RECOMMENDATION: Make new log for each contest, do not allow qrz/hamqth search. It slows things.

Note: This is **NOT** a contest logging add-on!!!
It is meant to be a tool for "Sunday contesters working in Tourist Class". Do not expect wonders!

Form Contest is just a "child form" for NewQSO to make qso logging faster when working in contests.

Form Contest has following fields:

- **Call** when you leave this field callsign is moved to NewQSO, to see if you have qso before.
- **RST s** copied from NewQSO, so should correspond used mode. Can be changed.
- **NR s** serial number to send. If **Inc** is checked it will auto increment. You may change this number at any time. Increment goes on from typed number.
- **MSG s** sometimes you may need an additional message with number, or just a message with no number. On high band contests you may type your locator here.
- **RST r** copied from NewQSO, so should correspond used mode. Can be changed.
- **NR r** serial number you receive.
- **MSG r** message you receive, if **MSG is LOC** is checked it will be placed **also** to NewQSO/Grid field.
- **Save QSO** "clicks" Save QSO button at NewQSO, increments NR and clears fields.
- **SPACE is TAB** when checked space bar acts like TAB-key moving to next field. Note! This prevents typing space (mainly to MSG fields).
- **Tru** checked menans you like to exchange true reports instead of 59/599
- **Inc** sets autoincrement to "Nr s" currently typed in when qso is saved.
- **No** changes Tab order so that "Nr r" is skipped (not used in this contest)
- **Clear all** clears all qso-related cells.
- **ESC key** returns cursor back to **end** of typed callsign. Pressing double ESC clears all.

TAB-key moves cursor between fields. So contest qso goes like this:
type callsign → press TAB → type contest rxnumber → press TAB → type contest message → press ENTER.
Or hit ENTER after rxnumber if no additional message is exchanged.

Remember that pressing ENTER key at any field saves qso with existing values. How ever there must be callsign at least three characters long. Sent/Received number and message are not needed.

Contest numbers and messages are saved in log into RST fields separated by spaces like. 599 001 MSG
This has two benefits:

- CW macros work without any programming change. If you set "send RST"-macro it sends also contest
  number and message, and when not in contest it sends just RST.
  How ever a small modification to original is made. If you use %rs for RST it will send 5NN. Now
  modified version sends also T for zero. So you will have 599 001 like 5NN TT1.

- Backup of logs work without any change. RST field just are a bit longer.

ADIF export is modified so that it separates RST fields to right tags.
<RST_SENT:3>599<STX:3>001
<RST_RCVD:3>599<SRX:3>002

<RST_SENT:3>599<STX:3>003<STX_STRING:2>SA
<RST_RCVD:3>599<SRX:3>003<SRX_STRING:2>dd

<RST_SENT:3>599<STX:3>002<STX_STRING:2>SA
<RST_RCVD:3>599<SRX:3>003<SRX_STRING:2>kb

<RST_SENT:3>599<STX:3>001<STX_STRING:2>SA
<RST_RCVD:3>599<SRX:3>002<SRX_STRING:2>SB

Freq	Mode	RST_S	RSTR
28	CW	599 001	599 002
21.0766	CW	599 003 SA	599 003 dd
21.0766	CW	599 002 SA	599 003 kb
21.0766	CW	599 001 SA	599 002 SB

Html export will look like this

Most contests expect Cabrillo log format. There is no support for this. You have to make ADIF export of qsos
and then use separate program if you want to send logs.

I found nice adif2cabrillo program for Linux from http://users.telenet.be/on4qz/ that supports ADIF importing.

Starting is bit complicated: you have to have qt5 and (qmake-qt5 if your distro is Fedora) to be able to compile
it.
Then there are no templates for different contests. You have to create them, but after that it serves you well!

It accepts nicely exported ADIF format from cqrlog and dig out STX and SRX as expected.

Change of ADIF-field name

If you want to use "Comment to qso"-field for adding special event info or something similar to your qsl
(eQSL) and cqrlog does have needed adif field for that you can easily change it afterwards.

To test this type **testing** to "comment to qso" and make adif **export** to file **qsos.adif.**
You will have "comment to qso" as: **<COMMENT:7>testing** in that file.
Open command terminal and type: **sed s/COMMENT/QSL_MSG/g qsos.adif > fixed_qsos.adif**
And you will have another file called **fixed_qsos.adif** with replacements as: **<QSL_MSG:7>testing** if the field
name "QSL_MSG" was the one you needed.

DXCluster Chat
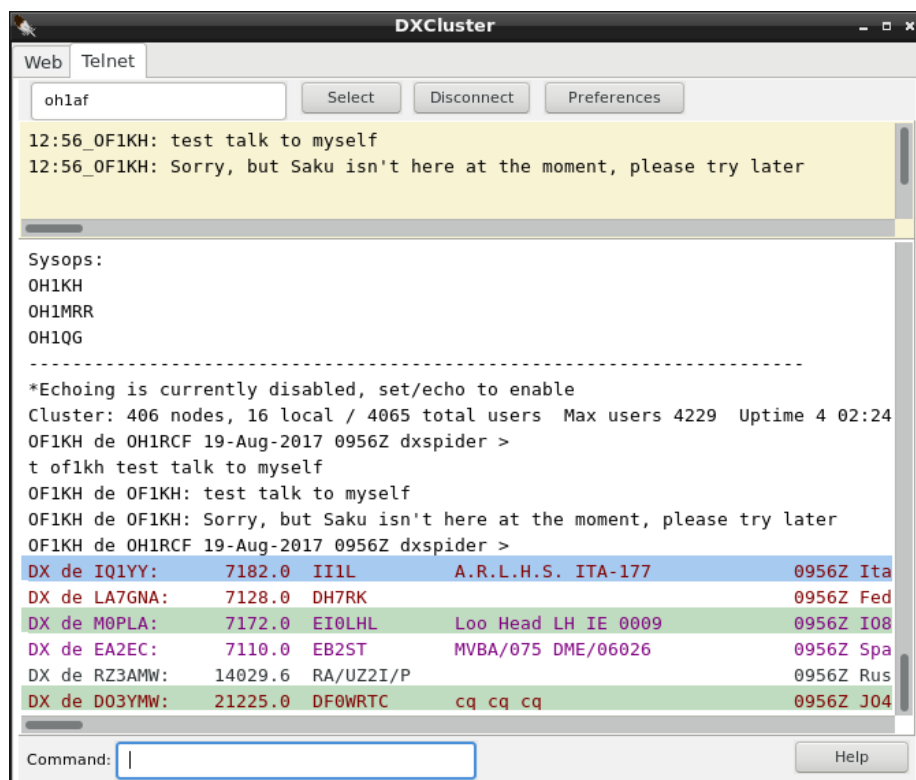
DXCluster window has now "overlay" as chat RX window. All lines that begins with "YOURCALL DE", but do not end to cluster prompt ">" are copied to this window.

Time(your local PC time) and caller's call are added in from of text line.

This works only in Telnet, not Web.

No more lost chats that are run over with the flood of DxSpots while you are visiting kitchen to get a cup of coffee.

DXChat uses the call you have used for logging in to DXCluster (that can be different as your current log settings call).

The size of Chat window, relative to Dx spot window can be adjusted pressing first button "Preferences" and then selecting "Chat size" from there.
After doing that "Command:"-cell at bottom of window is replaced with size slider.

Pressing the slider knob with mouse-left button and moving slider will reduce the size of chat window.
When mouse button is released "Command:"-cell returns and Chat window remains selected size.

If you have never used Chat window before default size will be 2 (=invisible).
Resize via "preferences"-button if you want to see it.
If you do not like this Chat window you can move/leave it to minimum size (2). Then it takes just horizontal 2px amount from DXSpot window.
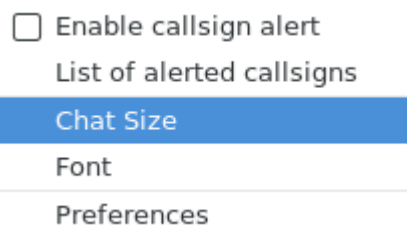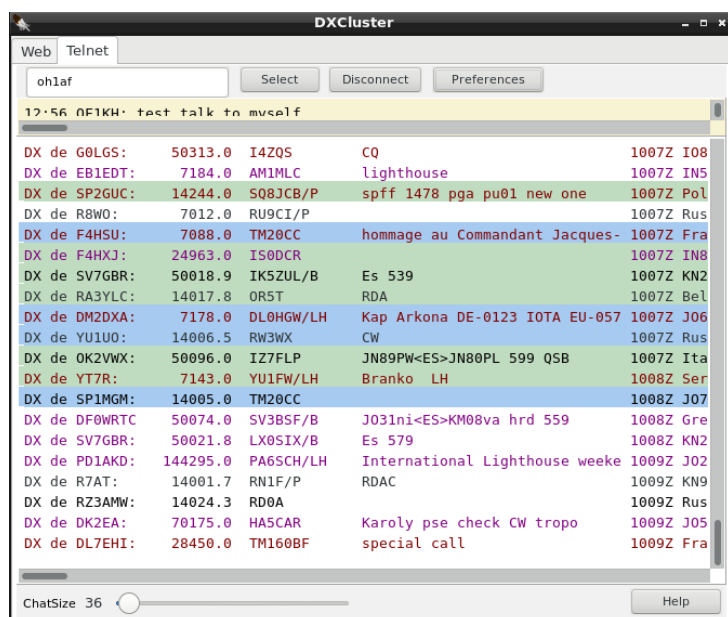
While minimized to up Chat window still receives all lines addressed to your call (and not having ">"-prompt at the end of line).
So any time you like you can make it larger and chat lines are there to check.

Band map filter

Setting from preferences/band map called "Show only active band" is **copied** also to band map filter window that opens from band map window/ band map filter button

Placing at preferences is fine when you are making settings for a new log, but for daily usage it is lot faster to find it from filters window.

And it actually is one filtering element, too.

ADIF and HTML export
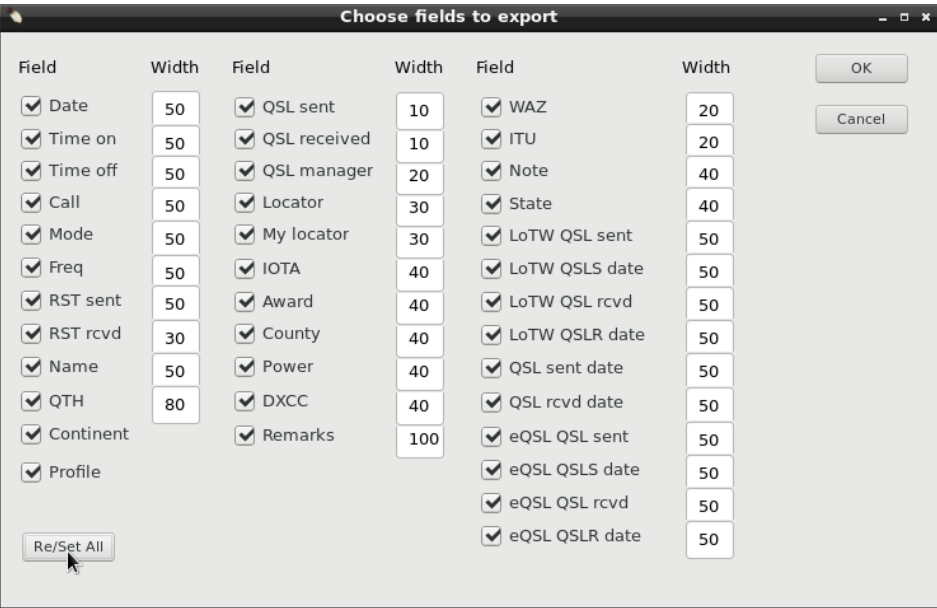
New button "Re/Set All" is added to "choose fields to export" window. That window you fill find when opening NewQSo/QSO list/File/Export after first entered file name where to export your log.

It allows fast set or reset all selections at one go. Every other click of button selects all, and next again deselects all check boxes. And so on.