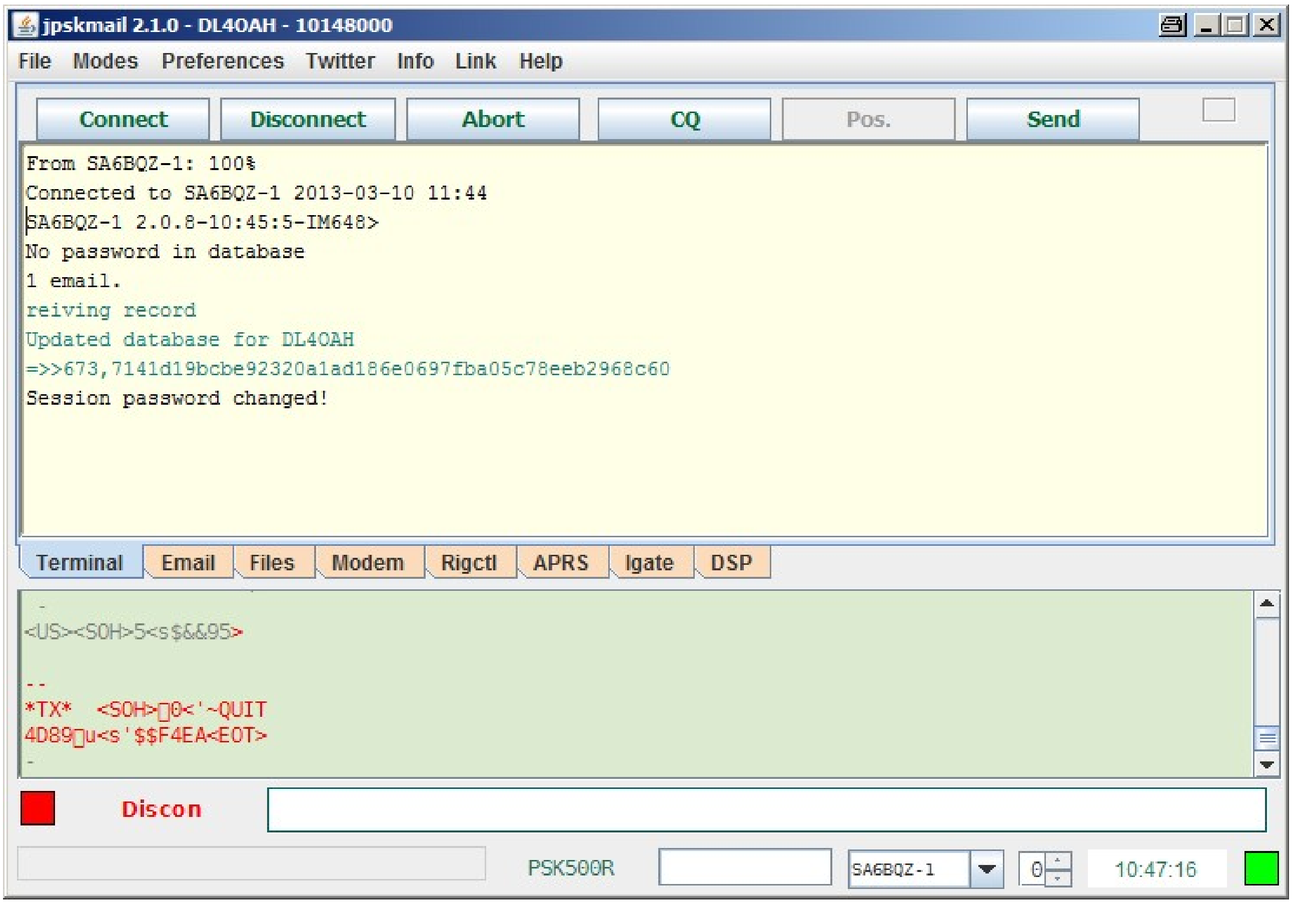
jPSKmail V2 user guide



```
From SA6BQZ-1: 100%
Connected to SA6BQZ-1 2013-03-10 11:44
SA6BQZ-1 2.0.8-10:45:5-IM648>
No password in database
1 email.
reiving record
Updated database for DL4OAH
=>>673,7141d19bcbe92320a1ad186e0697fba05c78eeb2968c60
Session password changed!
```

```
 -
<US><SOH>5<s$&&95>

--
*TX*  <SOH>□0<'~QUIT
4D89□u<s'$$F4EA<EOT>
-
```

PSK
mail

User manual updated March 15, 2013
Reflecting jPSKmail version 2.1.0

Contents

Preface

A brief attempt at writing the history of the project

The PSKmail project was started by Rein Couperus, PA0R early 2005. The baby steps were taken with a modification of gMFSK, the mode of choice was PSK63 and traffic was handled by the one and only server (PI4TUE) that existed. Soon another server joined the effort, SM0RWO in Stockholm, and suddenly the project had “multiple” users (two) and had to evolve. For a time there was only the server in Stockholm and two clients (PA0R and SM0RWO) and the project made necessary adjustments for the hf environment. Soon more servers started to appear, IS0GRB/Roberto, added a much needed server in the Mediterranean. Then more and more servers joined, such as WB5CON/Fred and DL9YCS/Jörg. The Intermar group then followed with several servers. At the start only email traffic was handled and, as stated earlier, only done so using PSK63. Soon more services were added, web browsing was one of those. One of the more important additions was APRS.

Traffic was handled nicely by PSK63 but yours truly, SM0RWO, felt the need for speed and noticed that the Fldigi project offered PSK125. Quickly Fldigi was integrated and PSK125 was the mode of choice for a long time. That urge for faster modes resulted in a move to PSK250 which is an excellent mode, it is of course faster and thus wider (abt 250 Hz). Wait a second, we almost always use 500 Hz filters in our rigs. Why not aim for a 500 Hz wide mode then? That mode was PSK500 and we moved there around summer of 2009.

Lengthy discussions within the group resulted in the consensus that it was time to break free from the chains of the old PSK31 keyboard to keyboard concept. PSK31 was indeed a fine idea but it had to make some sacrifices in order to reach its goal of being a suitable successor to rtty. Processing times could not be long, it had to deliver data pretty much in a similar way that the rtty operators were used to and one apparent problem was a lack of forward error control. A missing character here and there can mostly be handled by the human brain, not so by a computer with checksum control of received frames. VK2ETA, John Douyere, made it his task to create the more robust modes where FEC and an interleaver net was added to the PSK modes, the new mode is designated as PSKRobust 500/250/125 and has been a real success. In fact PSK robust 500 has been the default EU mode ever since.

Mode profiles were added by Rein, these control the server speed and mode that can be adapted by the server. The server can move a connected session to a faster mode when the conditions are right and can of course select a more robust mode when the conditions deteriorate.
While writing this, asymmetric linkup has been implemented. By that we refer to a connection that does not use the same mode and speed in both directions. For instance its now possible to have a weak client use a slow and robust mode while the server uses a very fast one. The mode scope now ranges from THOR4 to PSK500, using PSK, MFSK and THOR modes to cater for differing adverse conditions, including polar flutter, multipath, Doppler and QRN.

Version 2.0.x does away with an external modem. John and Rein ported the modems to java and integrated them into the application, which makes the configuration a lot easier.

What is PSKmail?

PSKmail is a system that uses data modes to enable clients to access both the internet and to communicate with other pskmail clients. Data modes are used over a suitable radio, mostly it's a hf radio but higher frequencies are also in use.

Just saying internet access could mean that the system should support a whole range of services, and indeed it does just that:

1. Messaging
    - Handles the users regular email account, including gmail etc.
    - For emergencies its possible to handle messaging off the internet (the whole internet could be down, messaging and local mail works anyway)
2. Web browsing, fetch and read any web page (text only)
3. Twitter, send and read tweets (via Tweetymail.com and Twitter.com)
4. File up and download with automatic resume at reconnect
5. Supports APRS including position reports, messages and short emails.
6. Supports map clients like Xastir and Uiview

But, that's just part of what's there now. Servers can also fetch your weather report periodically, show you where the nearest camp site is and tell what other users are nearby and so on...

So, what makes PSKmail special?

PSKmail is designed with the mobile/portable (maritime too) ham in mind. There is no big, power hungry and expensive modem to carry along. PSKmail is designed to use your PC soundcard as a modem.

A mobile/portable station is mostly not what we sometimes refer to as a “big gun”, antennas and power output tend to be on the modest side. With that in mind PSKmail uses bandwidth efficient modes, hardly ever wider than 500 Hz (for frequencies > 30 MHz that may of course be the case).

Wide data modes require a better signal to noise ratio than narrow ones, PSKmail is even able to adapt the speed and the mode so that the client can select a really narrow and efficient mode. Then again as the signal to noise ratio increases or the QRM goes away it will use wider and faster modes.

PSKmail network topology is flexible:

1. Clients can connect each other using ARQ, for chat or message transfer
2. Clients can connect to servers, which are gateways to the internet cloud
3. Clients can send each other messages in unconnected mode
4. Clients can send APRS message and position beacons to the APRS backbone through the servers
5. Clients also contain a passive Igate, which can gateway messages and posits to APRS
6. Clients contain a telnet port where you can connect a map application like Xastir or Uiview
7. The system also includes a telnet client to connect to telnet services on the internet or on a LAN connected to the server
8. Now that twitter is starting to be used for EMCOMM purposes, pskmail can generate tweets, just like your handy.

In summary, what is pskmail?

The main characteristics of PSKmail can be summed up as:

- A semi duplex ARQ Chat client capable of transferring files
- A full fledged solution for HF APRS, including integration with the map clients
- An internet connection that can be used where there is no internet connection. Such as from a boat at sea or a cottage somewhere or on expedition or during the next big storm...
- Narrow, efficient, data modes are used. Thus enabling even the most modest setup.
- Servers on short wave provide long range connectivity
- No expensive and bulky modem, all you need is a PC, a tablet or a smart phone, and a transceiver with an antenna
- Available on multiple operating systems (Linux and Windows, MacOS, FreeBSD and Android)

What about security in pskmail?

Starting with jPSKmail-1.2, the link is protected with a password. If no password is set, the link is insecure. To use the POP3 mail agent on the pskmail server it must know where to get your mail. Unlike other HF mail systems, the pskmail server does not store your mail, it has to get it from the mailbox at your ISP in the Cloud. The server uses a database to store your POP data, viz. Pop server, userid and pop password. The database can be updated from the client, by sending a record to the server with the 'update server' command. To make sure only the server gets the data the client sends the record using a cooky. By setting a link password via the Preferences dialog you make sure somebody else using your callsign is unable to get your mail. You have to do this only once for every server (unless the database gets lost...). You can also erase the record from the server.

Data (mail, files, web pages) are NOT encrypted on the link, but it is possible to send them compressed (zipped) to speed up transfers.

Installation and getting started

To use PSKmail version 2 you need:

1. **A ham radio license**
2. An HF transceiver with a working antenna solution, for boats that may include an antenna tuner and an insulated backstay or a whip antenna.
3. A computer running Linux, Mac OS or Windows. FreeBSD should also be fine but we lack test reports there.
4. An interface for connecting the computer soundcard to the HF transceiver. Some transceivers (like the K3 and the FT897) can be connected with a simple audio cable. Some modern transceivers even feature a built-in soundcard and CAT facility via USB...
5. A narrow filter (500 Hz) in the transceiver is highly recommended. An additional bandpass dsp may also be very nice but is normally not a replacement for a real filter.

To install and use the **jPSKmail client software** you will need additionally a j**ava runtime environment**. For Linux and MacOS you should be able to get that through your software distribution system (software update). For all others, windows, its here: http://java.com/en/download/manual.jsp

Install procedure

The install procedure is in principle the same for all operating systems, there are subtle differences and they will be shown further below. But, the steps are:

1. Make sure you have a **java runtime environment**
2. **Install jPSKmail using the installer program.** It is advisable to install in the suggested directory structure (jpskmail directory)
3. **Configure** the jPSKmail client using the Preferences->Edit menu.

Startparameter

The installer will install a start icon on your desktop. The standard command line is:

java -jar <ProgramInstallationFolder>javapskmail2.jar

Java will automatically use your local language and locale settings. These are not nearly fully implemented in jPSKmail. The language jPSKmail is written in is English. So you might see a mixture of your language and English.

The default name for your personal settings file is **configuration.xml**. If you want to do some experimenting and still keep your standard settings, you can add a filename as parameter. Example:

java **-Duser.country=EN -Duser.language=en** <ProgramInstallationFolder>javapskmail2.jar **MyConfig.xml**

Installation Instructions for RXTX

jPSKmail needs a JAVA Communication API to communicate with a GPS via the serial line (RS232 or USB), that library is the rxtx package. These steps are only necessary if you intend to actually use a GPS, or use hardware PTT. On Linux it is preferable to use **gpsd** as a GPS server.

Windows 98/NT/2000/XP/7

For installing the RXTX package you will find the necessary dll and jar files within the directory where you installed jPSKmail.

Installation procedure:

- copy rxtxSerial.dll to %JAVA_HOME%\bin,
  (%JAVA_HOME% is the folder where JRE is installed on your system; e.g. c:\Program Files\Java\j2re1.4.1_01)
- copy RXTXcomm.jar to %JAVA_HOME%\lib\ext

Linux

If you are using Ubuntu you may very easily take care of that by installing “librxtx-java” using synaptic or from a terminal with “sudo apt-get install librxtx-java”.

Others, manual procedure

For others, Linux other than Ubuntu, the simple procedure that follows involves copying files from the folder where jPSKmail was installed.

Installation procedure:

- copy librxtxSerial.so to <java-home-directory>/jre/lib/i386,
  (<java-home-directory> is the folder where JRE is installed on your system; e.g. /usr/local/j2sdk1.4.1_01)
- copy RXTXcomm.jar to <java-home-directory>/jre/lib/ext

Optional APRS mapping client

APRS contains geographical information such as positions of boats, vehicles etc. Its very nice to see that information on top of a map, that is why an aprs mapping client can connect to jpskmail and use that as a local internet server. JPSKmail acts as an APRS server for that client. To connect a map application to pskmail use tcp port 8063. The protocol is Xastir/UI-View compatible.

Hardware selection

Transceiver

There are many kinds of transceivers in use, many work really well with pskmail. Selecting a transceiver can be tricky and is of course guided by many factors. A few words of advice may be helpful for the beginner anyway.

A good transceiver should have an optimized digital mode, a mode where a narrow filter can be used for better reception. Adding a 500 Hz narrow filter is a very good way to enhance the S/N ratio on receive and may be the difference between a working and a non working connection.

Older transceivers may lack a dedicated digital mode and may require special handling in order to employ a narrow filter. One way around that is to use a split and listen in cw, with filter, and transmit in USB. That method may also require tweaking the rig BFO for cw. A memory for all of that is then useful.

As some of the modes (markedly the slower MFSK modes) used require precise TX and RX frequencies, choose a transceiver which is capable of doing that...

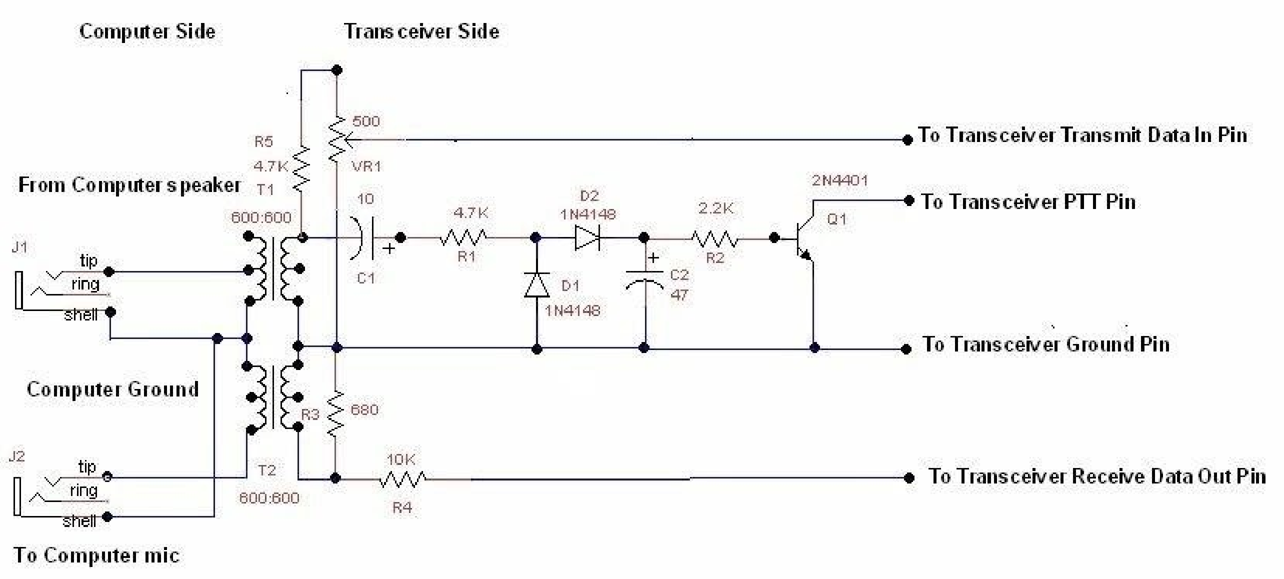
A server or client that intends to use some kind of scanning will require a computer control connector, such as CI-V or CAT or whatever hamlib will support. Computer control is not necessary for a client at the moment but could be useful for handling the PTT, and it enables summoning(TM).

Soundcard interface

The main difference between the version 1 and 2 clients is: version 1 requires the Fldigi program, version 2 contains modems written in java. They use the java sound system to interface with the soundcard in your computer. Your computer may have several sound cards like a built in and one ore more USB sound cards. ***jPSKmail does not contain audio drivers for your sound card, it can only use those sound drivers which are recognized by the java runtime system.***

The way java handles the audio is different for each operating system. jPSKmail provides combo boxes for choosing the input and output mixers. See description of the DSP Tab on page 14.

A soundcard interface is used to connect the computer to the transceiver. It may in theory be possible to connect straight lines between sound in/out and the microphone connector on the transceiver but that method has so many drawbacks that it is not recommended. A soundcard interface will provide isolation between the pc and the radio, it will normally be attached to the transceiver data connector and that helps keep the receive level correct. Please do consider using a soundcard interface. Here is an example of a simple audio/PTT interface from KH6TY:



The PSKmail protocol allows the use of VOX instead of PTT or soft PTT via CAT.

Some modern transceivers, like e.g. the IC7200, feature a built-in sound card and a CAT interface using a single USB connection. Which makes interfacing very simple. Other rigs like the K3 and the FT897 have built in audio transformers.

Antenna

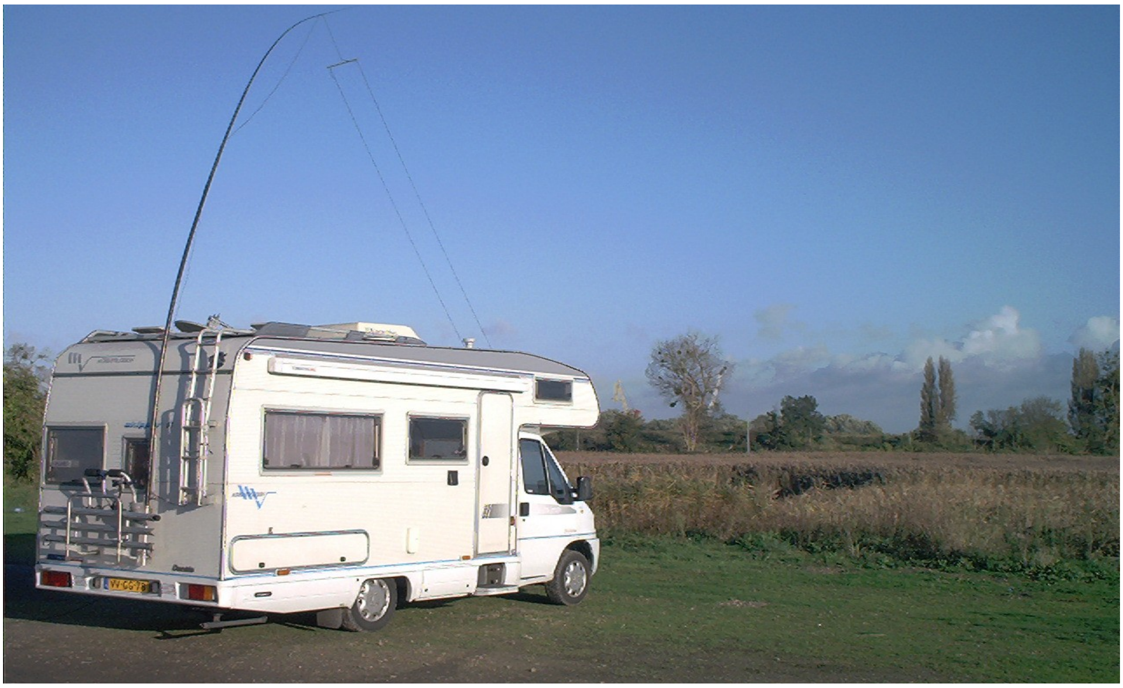
This is a favourite subject among radio amateurs and I suppose whatever I write here will be subject to discussion :-). Anyway, an old truth may help: big antennas are good antennas and small antennas are only small. You are more than welcome to discuss antenna choice on the pskmail mailing list, many experienced hams should be ready to help.

For a sailing boat a classical antenna is to use an insulated backstay with a tuner mounted close to that wire. Make shure the tuner is well grounded, e.g. connected to the keel. You will find heaps of information for mobile and maritime antenna systems in the net.

SM0RWO portable setup, from left:

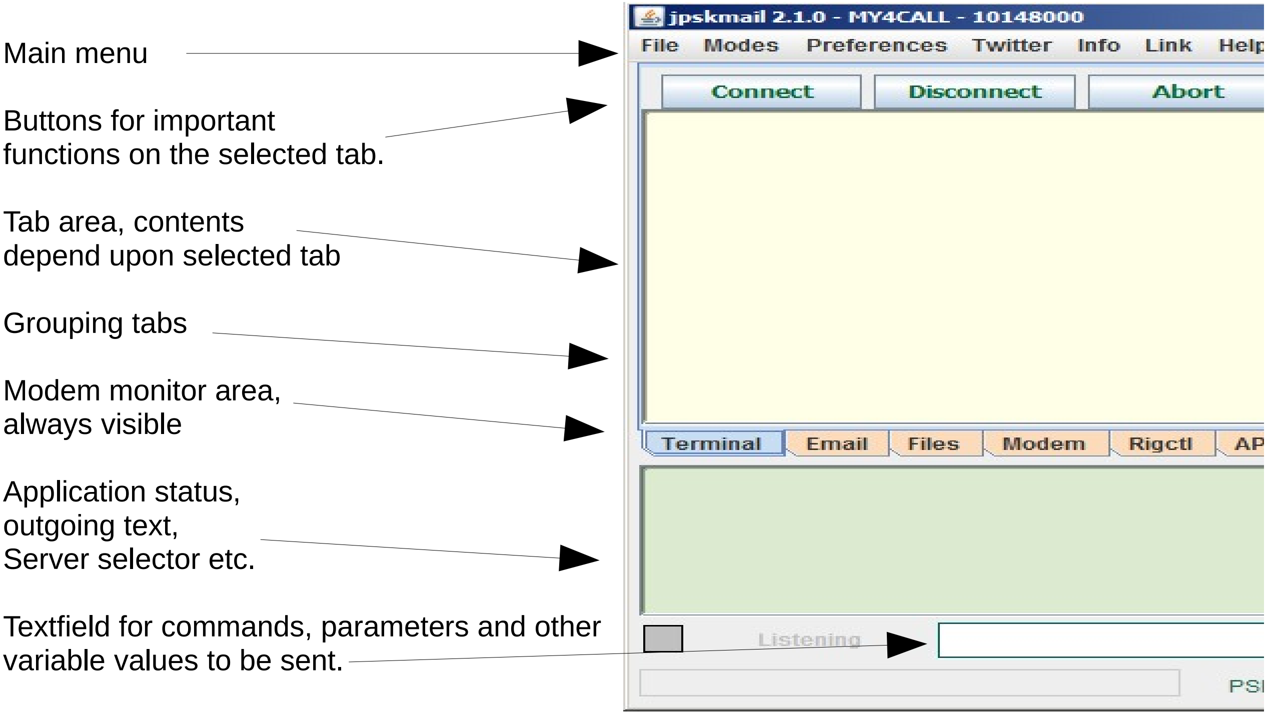
Yaesu FT-817-d, SB-2000 interface, EEE PC

An example of how to fit an antenna for the 80m band on top of an RV.



The Main Window

The Main window is vertically divided into several sections:



Main menu

Buttons for important functions on the selected tab.

Tab area, contents depend upon selected tab

Grouping tabs

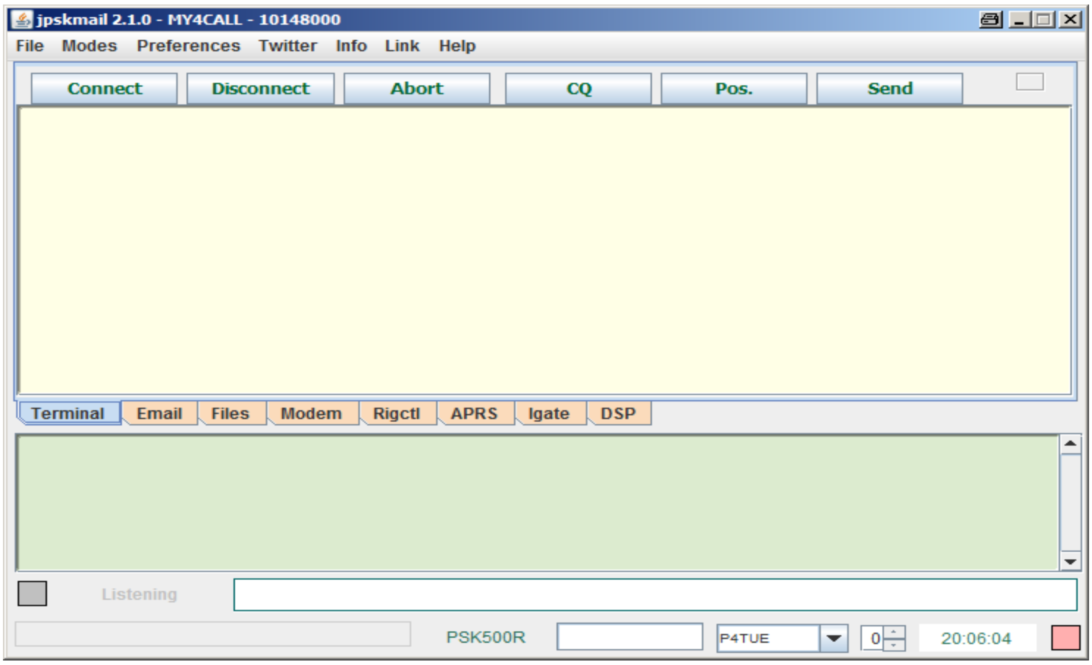
Modem monitor area, always visible

Application status, outgoing text, Server selector etc.

Textfield for commands, parameters and other variable values to be sent.

jpskmail 2.1.0 - MY4CALL - 10148000
File Modes Preferences Twitter Info Link Help
Connect
Disconnect
Abort
Terminal
Email
Files
Modem
Rigctl
AP
Listening
PSI

With the Tabs you can change the display for different tasks.



•**Menu bar**, a standard menu

•**Terminal**, shows the error corrected exchange with other stations.

•**Email**, where you can send and receive emails.

•**Files**, where you can upload and download files to a server or to another client.

•**Modem**, shows S/N, mode and lots of modem statistics, also includes statistics on what servers can be used.

•**Rigctl**, allows access to the scanner and summoning functions.

•**APRS**, the center of the APRS action including positions, messages, beacons etc.
Many will have that tab page up whenever not connected to a server.

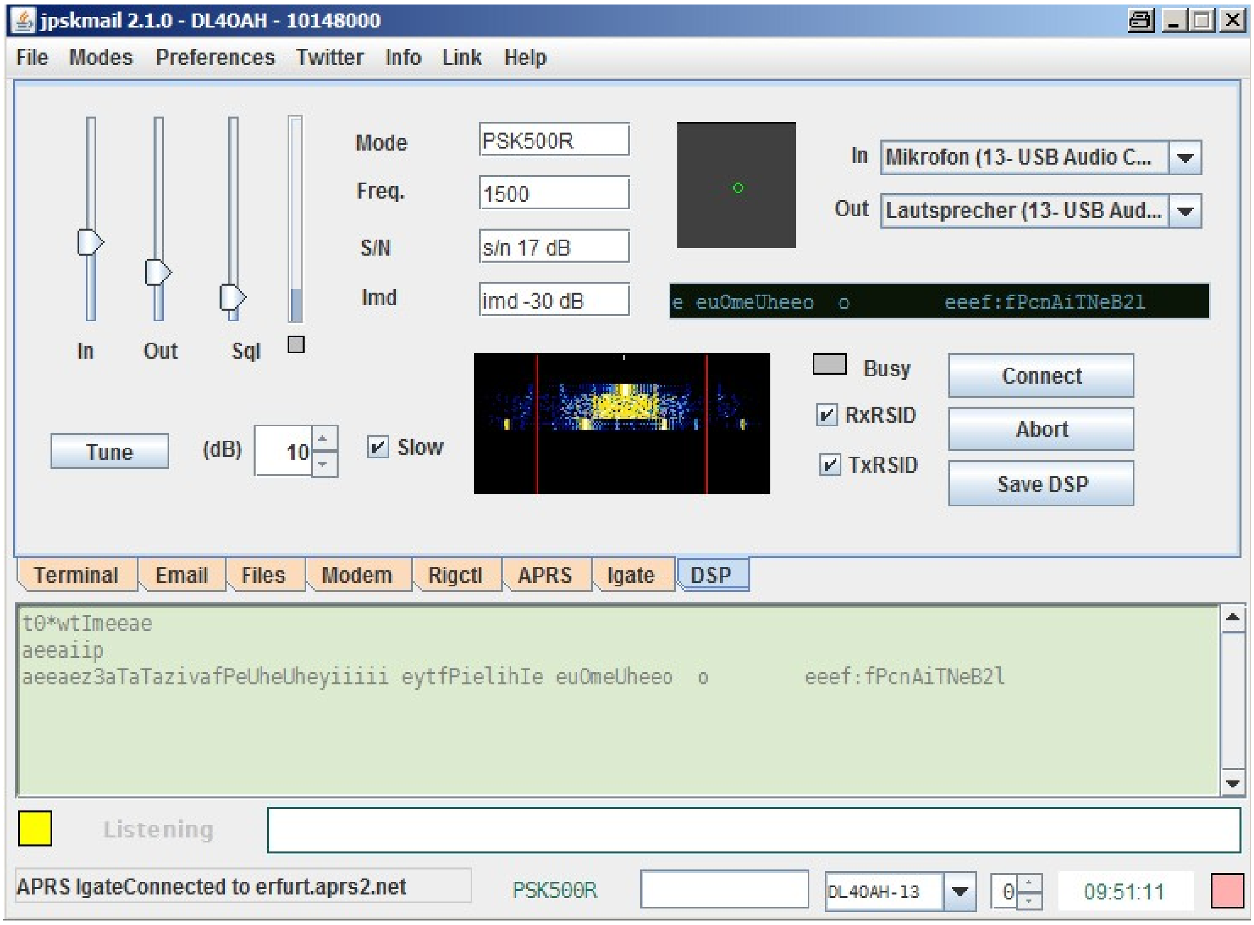
•**Igate**, shows all relevant information on what beacons have been relayed to the internet.

•**DSP**, shows controls for the interface to the audio system.

But, before going into detail and operation, some configuring must be done. Some of this is mandatory, like soundcard, own call and basic rig control. These are described first.

Configuration data are stored in a file with the default name **configuration.xml** in the folder **.pskmail** of your home directory. You can have different configuration files by using a parameter at program startup.
See page <TBD>.

DSP tab: Audio System (Sound Card)



jpskmail 2.1.0 - DL4OAH - 10148000

Mode: PSK500R

Freq.: 1500

S/N: s/n 17 dB

Imd: imd -30 dB

In: Mikrofon (13- USB Audio C...

Out: Lautsprecher (13- USB Aud...

(dB): 10

☑ Slow

☑ RxRSID

☑ TxRSID

```
e euOmeUheeo  o          eeef:fPcnAiTNeB2l
```

```
t0*wtImeeae
aeeaiip
aeeaez3aTaTazivafPeUheUheyiiiii eytfPielihIe euOmeUheeo  o        eeef:fPcnAiTNeB2l
```

APRS IgateConnected to erfurt.aprs2.net PSK500R DL4OAH-13 0 09:51:11

The DSP tab is used for control and display of the interface to the sound system.

- **In**

  Slider to control the input level. See also sound card handling below.

- **Out**

  Slider to control the output level. See also sound card handling below.

- **Sql**

  Slider to control the squelch level.

- **Level**

  Display of signal level.

- **Tune**

  Send an UI Block for tuning purpose like setting output level to a value where the transceivers ALC just starts to operate.

- **(db)**

  Additional input attaenuator 0 – 30 dB in 10 dB steps.

- **Slow**

Slow down waterfall display a bit if checked.

- **Mode**

  Show the current mode.
- **Freq.**

  Show the current offset frequency, the “sweetspot”.
- **S/N**

  Show the current Signal to Noise ratio.
- **Imd**

  Show the current intermodulation value.
- **Busy**

  Indicate a busy state.
- **RxRSID**

  Use incoming RSID when applicable.
- **TxRSID**

  Use incoming RSID when applicable.
- **Connect**
- Initiate a connect, same as **Connect** on the **Terminal** tab or **<ALT-C>**.
- **Abort**
- Initiate an abort, same as **Abort** on the **Terminal** tab or **<ALT-A>**.
- **Save DSP**

  Click this button to make settings on this tab permanent. **Don't forget this!**
- **In**

  Select your input sound channel (mixer). See below.
- **Out**

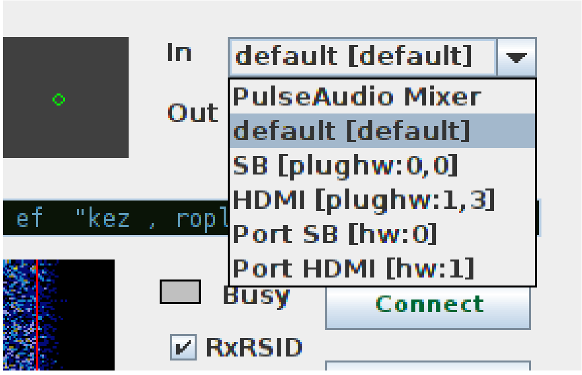
  Select your output sound channel (mixer). See below.

The waterfall displays the received spectrum, low to high frequency from left to right, sweet spot frequency in the center. Black to dark blue color is low, white is high amplitude.

The DSP tab also contains a phase indicator for PSK modes.

Basically, the audio levels should be mid-range, and the squelch at 10%. **The audio should be adjusted with the audio settings panel of your OS, in such a way that the waterfall noise background is blue, and the signals yellow/white.**

jPSKmail audio on Linux



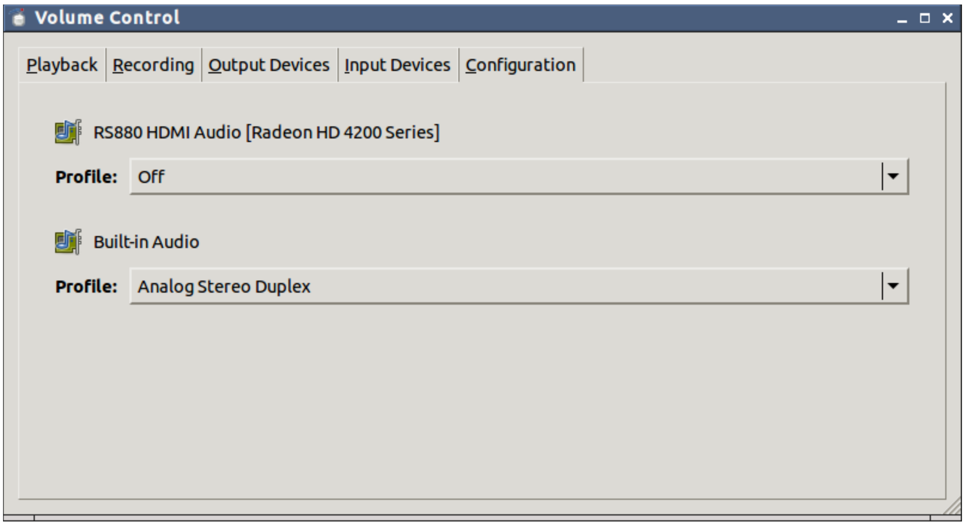
The list of mixers will be different depending on which OS you use. In the upper screenshot its a german windows 7, to the left you see LUBUNTU 12.04, with pulse audio doing the work.

The easy way is to set up the default audio mixers with the audio panel on your machine first, and then choosing the default mixer on the DSP tab.

But first set up the audio mixers:

Here is what it looks like on the LUBUNTU system using the **pulseaudio volume control:**

Configuration:



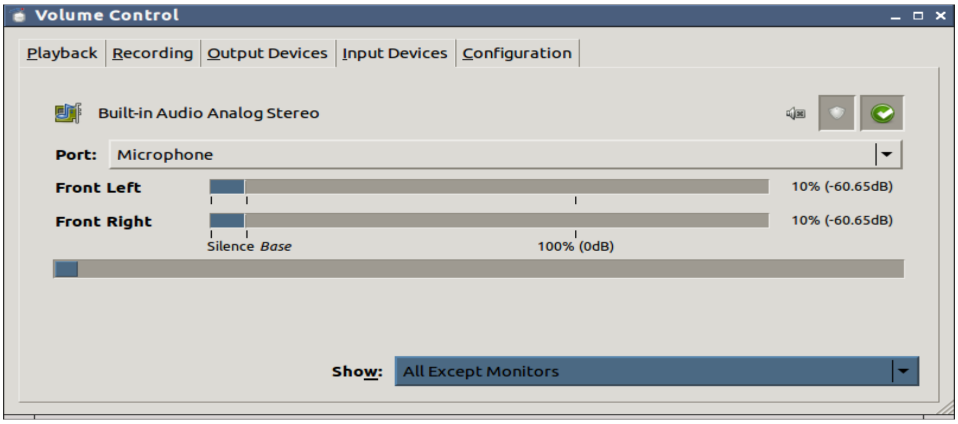
RS880 HDMI Audio [Radeon HD 4200 Series]

Profile: Off

Built-in Audio

Profile: Analog Stereo Duplex

Input Devices:



Built-in Audio Analog Stereo

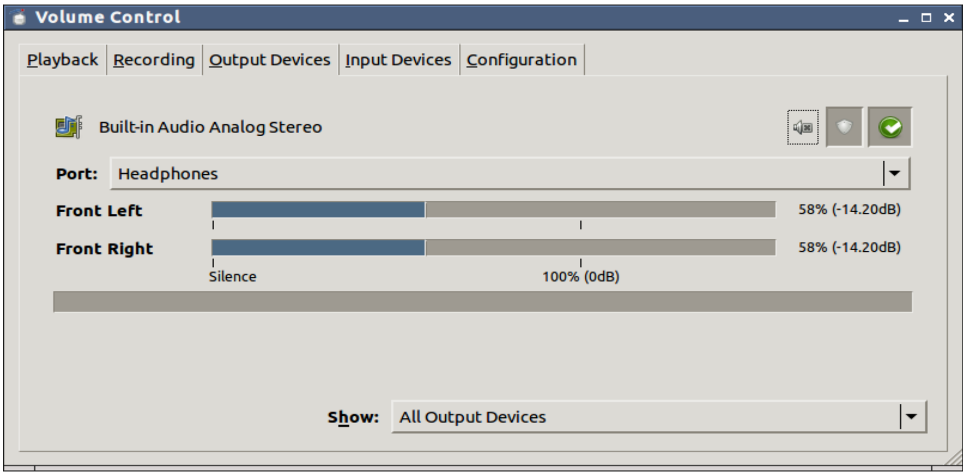
Port: Microphone

Front Left 10% (-60.65dB)

Front Right 10% (-60.65dB)

Show: All Except Monitors

Output devices:



Built-in Audio Analog Stereo

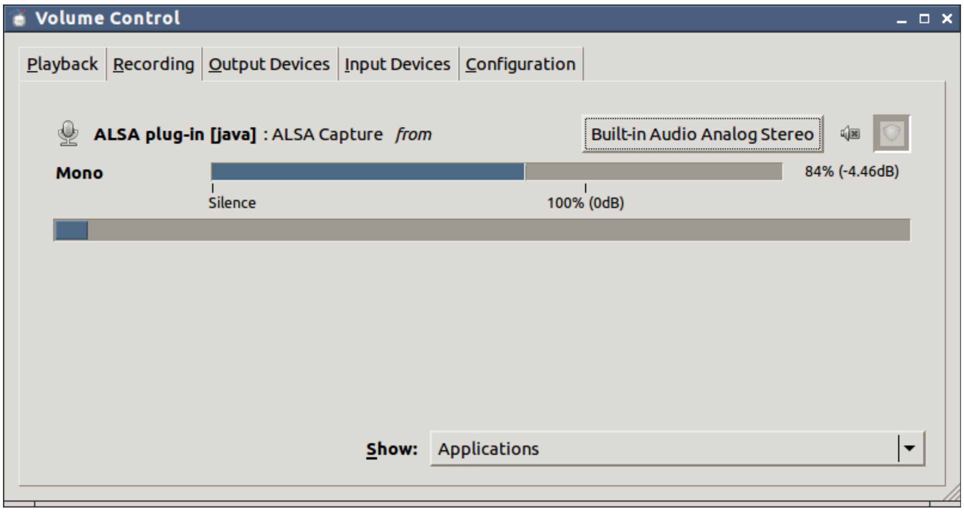
Port: Headphones

Front Left 58% (-14.20dB)

Front Right 58% (-14.20dB)

Show: All Output Devices

Capture:

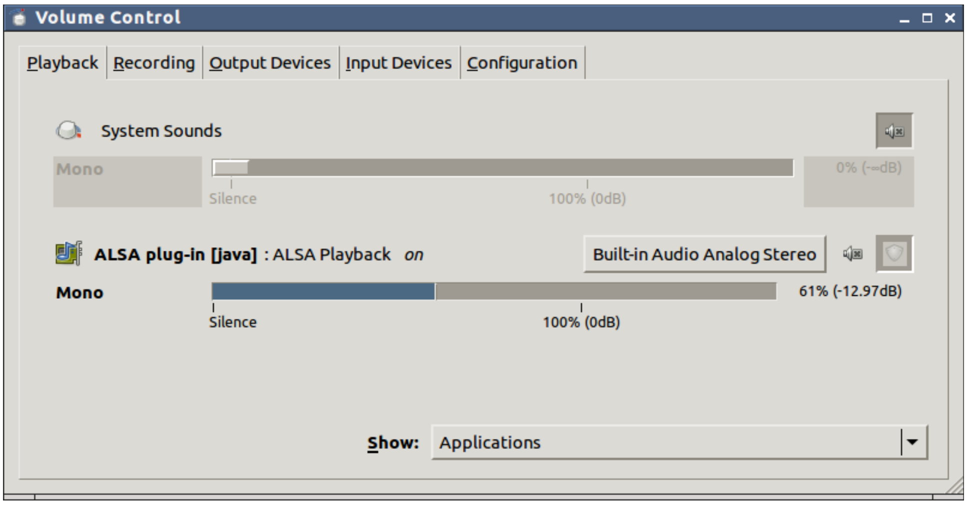


ALSA plug-in [java] : ALSA Capture *from* Built-in Audio Analog Stereo

Mono 84% (-4.46dB)

Show: Applications

Playback:



System Sounds

Mono 0% (-∞dB)

ALSA plug-in [java] : ALSA Playback *on* Built-in Audio Analog Stereo

Mono 61% (-12.97dB)

Show: Applications

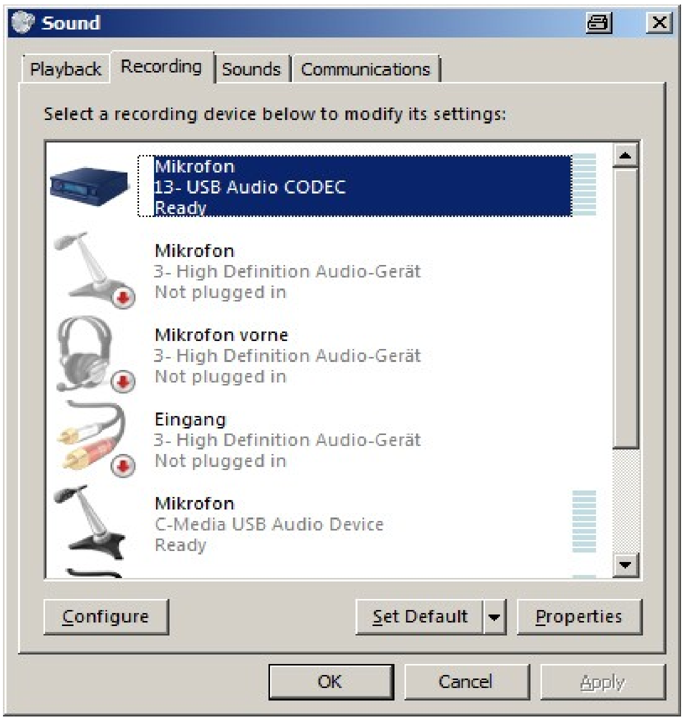
When you use pulseaudio on Linux, leave the pavucontrol running!

The above shows the basic principle of setting up audio on jPSKmail version2. On other operating systems the basic principle will be the same, **it just looks different. I have tested audio configuration on various Linuxes, WIN XP, and WIN7.**

jPSKmail audio on Windows 7

First plug in the USB soundcard or the USB cable from tranceiver or from your interface module. Of course you can also use a built-in or PCI-based soundcard..

The easiest way to change settings is RMK (Right Mouse Key click) on the speaker symbol in the task bar, then select **Playback** or **Recording** devices.



Select a recording device below to modify its settings:

Mikrofon
13- USB Audio CODEC
Ready

Mikrofon
3- High Definition Audio-Gerät
Not plugged in

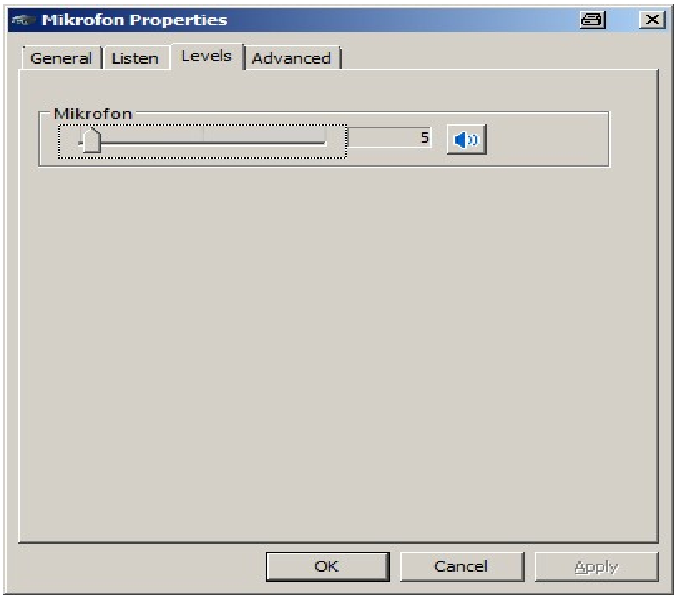
Mikrofon vorne
3- High Definition Audio-Gerät
Not plugged in

Eingang
3- High Definition Audio-Gerät
Not plugged in

Mikrofon
C-Media USB Audio Device
Ready

This is the recording Tab. The checked item (… 13-USB-Audio-CODEC...) is an interface module with builtin USB sound card, connection to the tranceiver. You find the same name in the **In** selection control of DSP Tab.

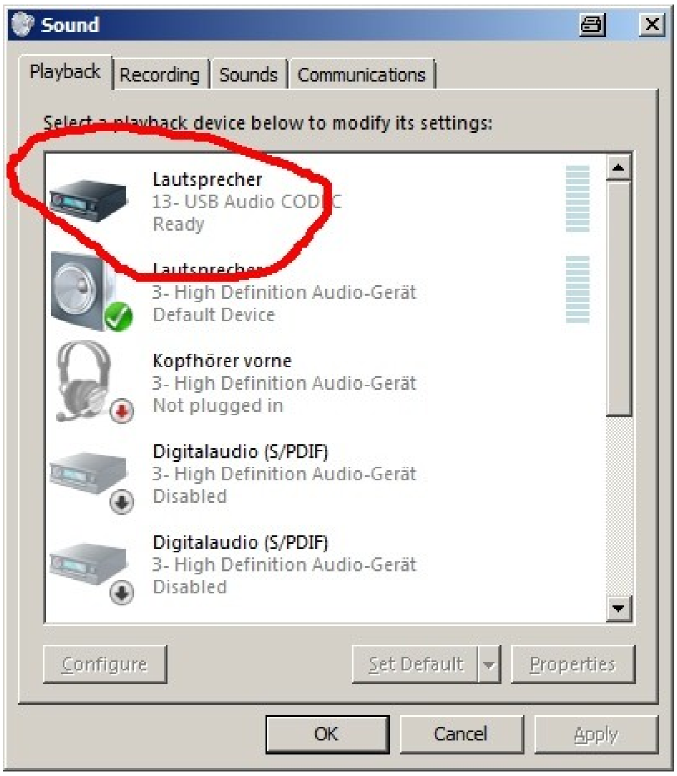
Click the **Properties** button.



Mikrofon 5

Select **Levels** on the next window and you have a slider to set the input level. It should be set to a value where the slider on DSP Panel is in the middle and you can do the fine adjustment there.

Handling of output channel is similar.



Select a playback device below to modify its settings:

Lautsprecher
13- USB Audio CODEC
Ready

Lautsprecher
3- High Definition Audio-Gerät
Default Device

Kopfhörer vorne
3- High Definition Audio-Gerät
Not plugged in

Digitalaudio (S/PDIF)
3- High Definition Audio-Gerät
Disabled

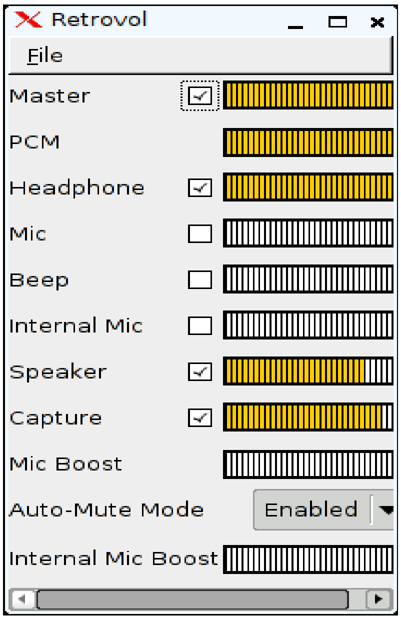
Digitalaudio (S/PDIF)
3- High Definition Audio-Gerät
Disabled

This is the playback Tab. The checked line is the standard system speaker. The marked first item (… 13-USB-Audio-CODEC...) is the interface module to transceiver with builtin USB sound card. You find the same name in the **Out** selection control of DSP Tab.

Click on the desired device and then the **Properties** button. Select **Levels** on the next window and you have a slider to set the output level. It should be set to a value where the slider on DSP Panel is in the middle and ALC of your transceiver just starts to react.

Do not overdrive the transceiver! Higher power is not necessarily higher quality.

This is what it looks like on puppy Linux 5.4.3:



Master ☑
PCM
Headphone ☑
Mic ☐
Beep ☐
Internal Mic ☐
Speaker ☑
Capture ☑
Mic Boost
Auto-Mute Mode Enabled
Internal Mic Boost

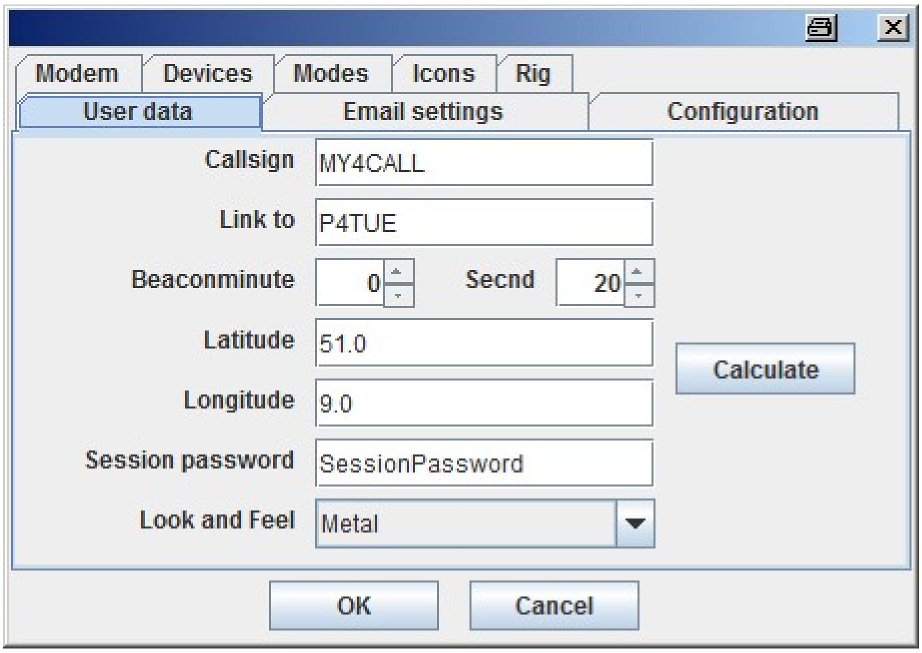
After OS setup select the sound channels, also called Mixer, you want to use on the DSP Tab and click the **Save** button. Possibly you have to restart jPSKmail to see the waterfall.

Preferences Setup (Options)

After the first sound setting a few basic things are still left. For this click on **Preferences** on the Main Menu and then on **Edit** or simply hit **<CTRL-E>**.

You get a new window showing a lot of options on several Tabs. Two settings are required before we try a first start “on the air”.

User data



Callsign MY4CALL

Link to P4TUE

Beaconminute 0 Secnd 20

Latitude 51.0

Longitude 9.0

Session password SessionPassword

Look and Feel Metal

1. **Callsign** Your call. **This is mandatory!**
   Please use **CAPITALS** for your call, and don't use additions like ZL/ or /M.
   SSID's (-2, -15) are allowed.

   Your email settings are stored at the server using the callsign as key. If you wish to check several email accounts then you could tie them to an ssid each. Like no ssid for primary email account and -2 for gmail. Just type in the first, update the server and repeat the procedure for the next account.

2. **Link to**
   Call of the server or client you want to connect, normally the one you can hear best.

   The monitor tab shows you the quality of the beacons received by the client.

   There is a list of servers on the pskmail wiki, server operators are requested to keep that list up to date: http://pskmail.wikispaces.com/PSKmailservers

   The **server call** can be changed at runtime with the server selector at the bottom right of the main screen. See page 46.

   On program start the server is loaded from the configuration file. Your jPSKmail client will automatically add more servers to the drop down list as QSL's of beacons and messages are received. The list is editable, you may add a server call any time at runtime. But with every program start the server selector will contain only the call you enter here.

3. **Baeconminute**
   The minute the client beacons. The server has 1 minute time slots with a 5 minute period. The beacon minute runs from 0...4. It is used to make sure you don't TX on top of someone else. Also some servers are scanning several frequencies, and you can choose the channel for the beacon this way.

4. **Secnd**

   For fine tuning of the beacon timing. Beaconing will try to start at this second within the minute.

5. **Latitude**
   In decimal degrees (e.g. 40.057365) in case you do not use a GPS unit. You may leave this field empty if you connect a GPS unit. North is positive, South negative.

6. **Longitude**
   In decimal degrees (e.g. -0.072185) in case you do not use a GPS unit. You may leave this field empty if you connect a GPS. East is positive, West negative.

   0/0 position is **not** sent in APRS beaconing. The calculate button raises a self-explanatory window where you can calculate between the different Position formats, use a current GPS position if available, transfer the result to the fields on this User data Tab. Different position formats are:

   Decimal: -DDD.dddd like -152.1234 for a western longitude near 152°

   APRS: DDDMM.mmN like 5201.18N for a northern latitude near 52°01'

   Seconds: DDDMMSS like 11°22'33''E for a eastern longitude

7. **Session password**

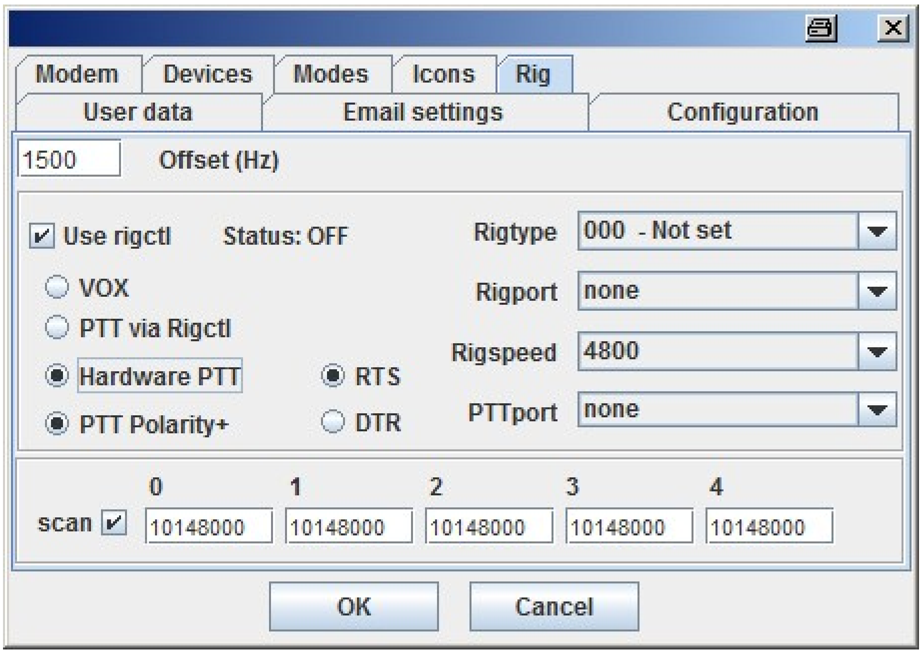
   Used to prevent misusage of your call.

8. **Look and Feel**

   A combo box to choose the appearance of the program.

   Default is the operation system standard, **Metal** is pretty similar between the os'ses. Not all settings are useful in all environments.

Rig



1500 Offset (Hz)

☑ Use rigctl Status: OFF

○ VOX

○ PTT via Rigctl

● Hardware PTT ● RTS

● PTT Polarity+ ○ DTR

Rigtype: 000 - Not set

Rigport: none

Rigspeed: 4800

PTTport: none

scan ☑

0	1	2	3	4
10148000	10148000	10148000	10148000	10148000

1. **Offset**

   Here you set the **sweetspot** of your transceiver. This means the middle frequency of your filter settings. On most rigs the correct value is 1000 Hz or 1500 Hz. Elecraft K3, icom IC-7000 and IC-7200 must be set to 1500.

2. **Use rigctl**

   If you leave this checkbox OFF, the transceiver must be capable of VOX control in your setup. Often this is not the case, when you use the transceivers digital facilities. It also means, that you cannot use scanning and summoning.

3. **VOX**

   VOX, PTT via Rigctl and Hardware PTT form a radio button set. With rig control you may still use VOX for receive/Transmit switching.

4. **PTT via Rigctl**

   Use this setting if you want frequency setting and soft PTT via a serial Port. You must select correct values for rigtype and serial port. Not all rigs with CAT capabilities can handle soft PTT.

5. **Rigtype**

   Select your Rigtype here. The list is derived from the **rigctl program** (hamlib project), which must be installed if you want to use rig control. See page 63.

6. **Rigport**

   Select the serial port for rig control here or **none**. The program offers all ports it could identify. It is up to you to choose the correct port, which is wired for this purpose and not used by another function of this program or another program running at the same time.

7. **Rigspeed**

   Select the speed (baudrate) for the chosen port. Other port settings are fixed to 8 bit, 1 stopbit, no parity.

8. **Hardware PTT**

   Use this setting if you want to use PTT via a hardware signal.

9. **PTTport**

   Select the serial port for hardware PTT or **none**. This Port must be different from **Rigport** above and from GPS port if you select one on the **devices** Tab. Only DTR **or** RTS pin of this port are used.

10. **PTT Polarity+**

    Determines the polarity of hardware PTT pin. Checked means nominal +12V during transmit, unchecked nominal -12V.

11. **RTS**

    Checked tells the program to use RTS for Hardware PTT. RTS is pin 7 on 9-pin RS232 and pin 4 on 25-pin RS232.

12. **DTR**

    Checked tells the program to use DTR for Hardware PTT. DTR is pin 4 on 9-pin RS232 and pin 20 on 25-pin RS232.

13. **Scan**

    Select Scan to use the scanning capability of jPSKmail. This is only possible with a working rig control.

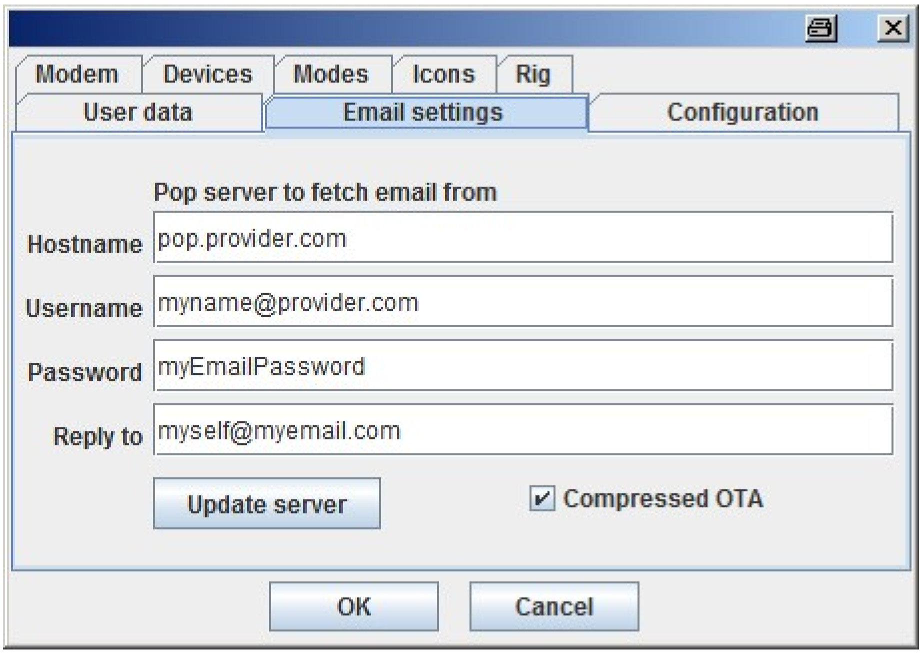
14. **scan frequencies 0-4**

    Here you set the desired nominal scanned frequencies for every minute of a 5 minute cycle. Be careful to have appropriate antenna or tuning setup if you transmit on different frequencies. During scanning the transceiver will show this frequency **minus Offset** (sweetspot, see item 1).

You can have a specific scan table for every server. Just put a file with SERVERCALL.chn into the .pskmail directory containing a list of 5 frequencies, comma separated. The list of frequencies of scanning and not scanning servers can be found on:
http://pskmail.wikispaces.com/PSKmailservers

Now we have the basic settings. But there are more:

Email settings



Pop server to fetch email from

Hostname: pop.provider.com

Username: myname@provider.com

Password: myEmailPassword

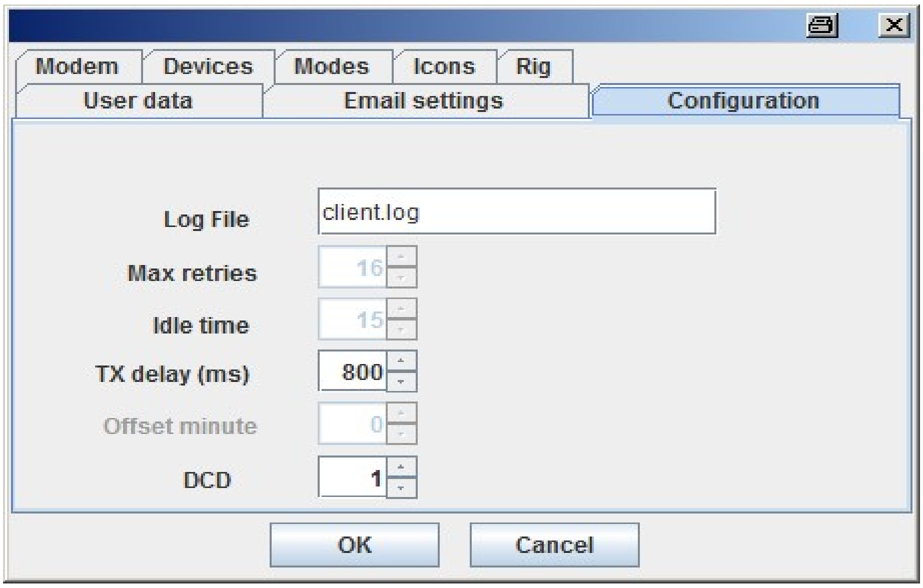
Reply to: myself@myemail.com

☑ Compressed OTA

1. **Hostname**
   The name of your pop server, e.g. pop.myisp.com. If you intend to use a gmail account then enter pop.gmail.com here. T**he name is different for every ISP, visit their website for info**.
2. **Username**
   Your POP userid. For gmail it is your full account name (including @gmail.com or @your_domain.com).
3. **Password**
   Your POP password, related to POP username.
4. **Reply to**
   When someone wants to respond to your email then the reply will be sent to this address. If you want to receive the answer when away then use the address you can check. When sending emails an SMTP server checks whether your return address is valid. This is also necessary for sending one-line emails via APRS. **A server which does not have your return address may be unable to send your email.** For gmail this is your full account name (including @gmail.com or @your_domain.com).
   It is now also possible to use a different return address for every mail, which is convenient when e.g. you are on a boat with several people
5. **Update Server**

   This button sends your data to the server you are connected to.
6. **Compressed OTA**
   Check this to enable compressed email handling and web page downloads over the air. Compressed means that emails and web pages are transferred as binary zipped files, this can speed up the transfer by a factor of 2x.

Configuration



Log File client.log

Max retries 16

Idle time 15

TX delay (ms) 800

Offset minute 0

DCD 1

In the Configuration preferences you can set some important running parameters which fix the protocol timing etc.

1. **Logfile**

   Name of the file where the program logs some events and error messages. The file is located in your home directory in the folder .pskmail. Normally the is no need to change the default name.

2. **Max retries**

   Disabled

3. **Idle time**

   Disabled

4. **TX delay (ms)**

   The TX delay sets the time in milliseconds the client waits for the server to finish its last frame.

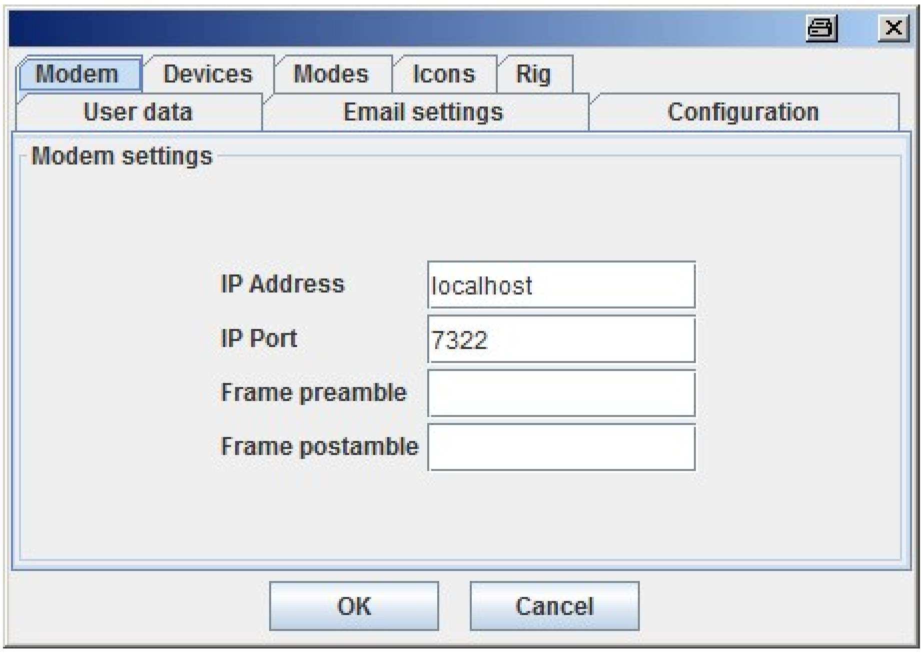
5. **Offset minute**

   Disabled

6. **DCD**

   The DCD value is important for the channel-busy function. A value of 1 or 2 seconds is appropriate, a value of 0 switches DCD off.

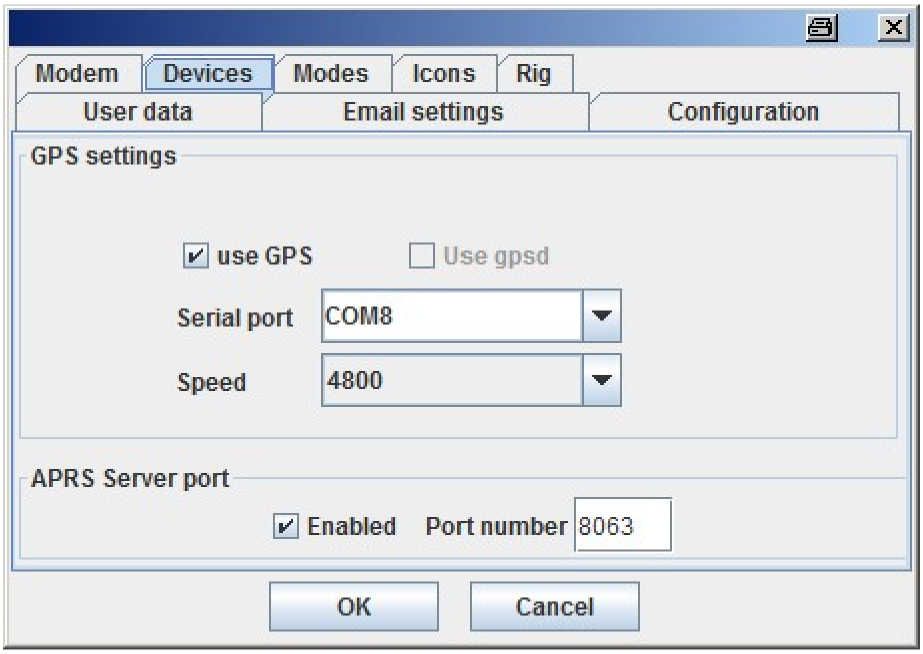
Modem



IP Address: localhost
IP Port: 7322
Frame preamble:
Frame postamble:

The Modem settings are not used with client 2, but this may change in the future.

Devices



GPS settings

☑ use GPS   □ Use gpsd

Serial port: COM8

Speed: 4800

APRS Server port

☑ Enabled   Port number: 8063

This tab controls external devices and connections. At the moment that means an external GPS and the internal APRS server for external mapping clients. The GPS is used for updated position beaconing and also GPS time is fetched and shown on the client. The internal APRS server is used from an aprs mapping client, for instance xastir. APRS data is fed to the mapping client so that geographical aprs information can be viewed on top of a map.

1. **Use GPS**
   Check this if you want to use GPS. Either a serial port or, only on Linux systems, a working gpsd deamon is required.

2. **Use gpsd**

   gpsd is a gps data server which is only available in Linux. It allows several programs to use the data of the gps unit. jPSKmail uses the new gpsd protocol, so you must use gpsd version > 2.9.0.

3. **Serial port**

   Serial port where the GPS is connected or none. See serial ports on the rig tab.

4. **Speed**
   Standard NMEA speed is usually 4800 bps but if other speeds are used then just select the correct one here.

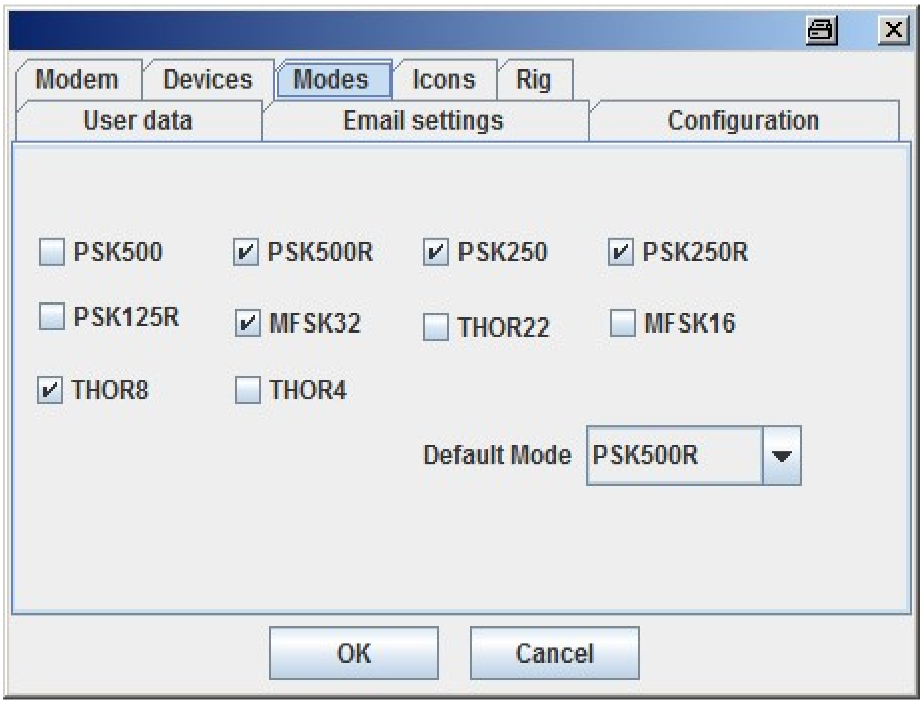
5. **APRS Server port enabled**

   jPSKmail can act as your own internet APRS server. Just point an APRS mapping client to localhost. jPSKmail and the mapping client must reside on the same computer.

6. **Port number**

   Connect the APRS client to **localhost** and the port noted here.

Modes



□ PSK500 ☑ PSK500R ☑ PSK250 ☑ PSK250R

□ PSK125R ☑ MFSK32 □ THOR22 □ MFSK16

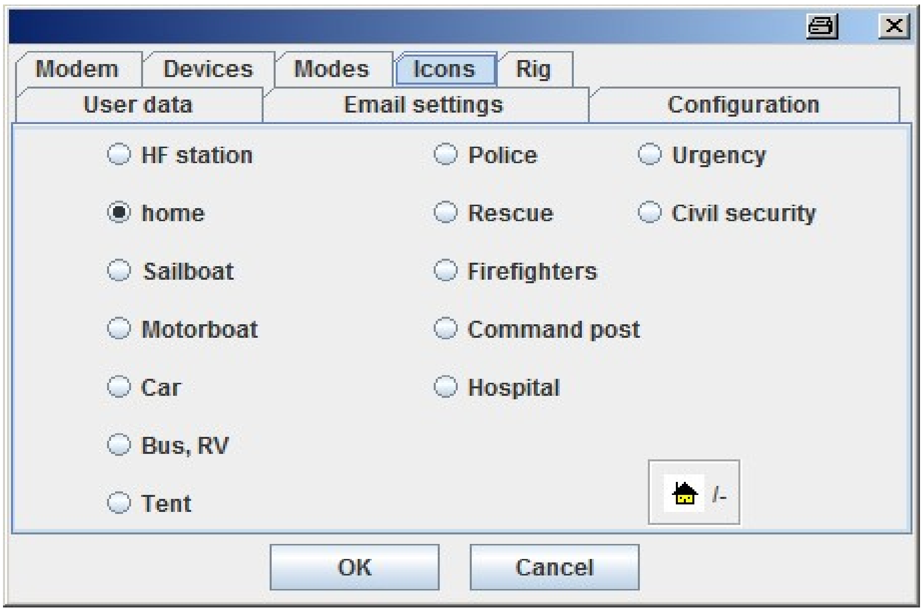
☑ THOR8 □ THOR4

Default Mode PSK500R

Although there is a reasonable default setting you can select a list of modes you want the client to use. This tab shows the implemented modes in descending order of speed, which implies in most cases an ascending order of robustness. Prepare your own list by checking or unchecking the values. Also select your **Default Mode**, which is normally **PSK500R** and **PSK250** in the US.

During a connected session the server will switch the modes up and down this list, depending on the channel conditions. When the client is not connected it will switch to the **Default Mode** automatically. It does this on the top of every minute.

Icons



○ HF station
● home
○ Sailboat
○ Motorboat
○ Car
○ Bus, RV
○ Tent
○ Police
○ Rescue
○ Firefighters
○ Command post
○ Hospital
○ Urgency
○ Civil security

On this tab you find a selection of APRS icons. Choose the one you want to be used in your beacons. The corresponding graphical and APRS internal representation is shown here and on the main APRS tab.

Store settings

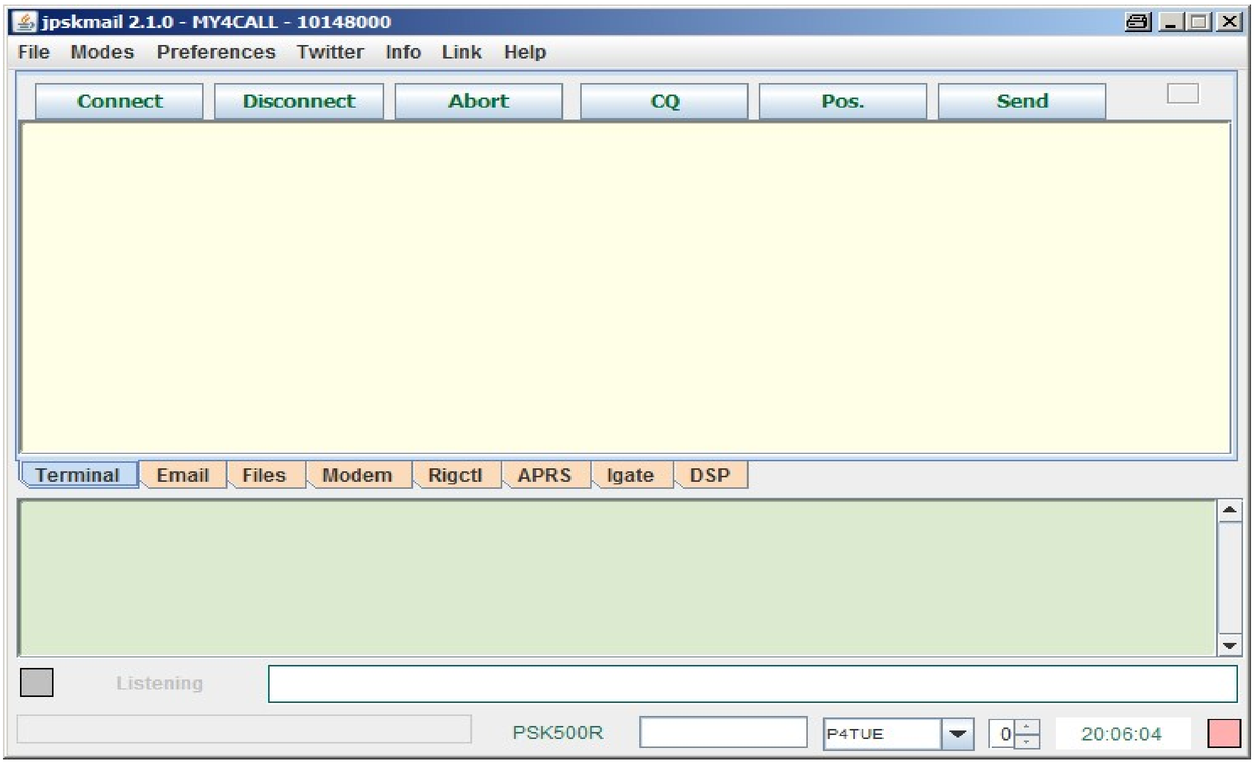
Click **Cancel** to abandon your changes, **OK** to store them. You may also use the **<CTRL-S>** key, which will store all settings: Preferences, DSP, APRS and Igate.

User interface tour

The jPSKmail application has an interface with some permanently visible parts and some that are grouped and accessible through tabs. Upon starting the application it will display the main window.

Terminal tab

Connected operation (ARQ)

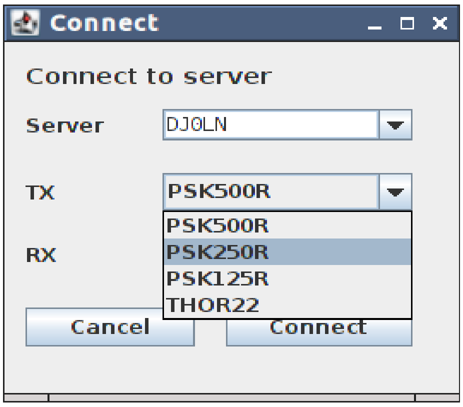


jpskmail 2.1.0 - MY4CALL - 10148000

PSK500R P4TUE 0 20:06:04

On the terminal tab you can start a connected session with the server.

1. Hit the **Connect** Button or go to Files->Connections->Connect
   The Connect button will give you a connect window where you can choose the modes.



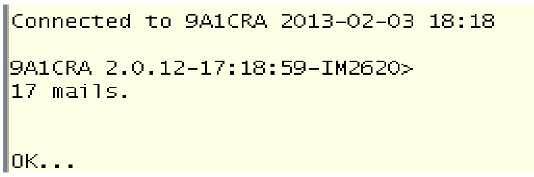
Connect to server

Server DJ0LN

TX PSK500R

RX

2. Wait until server acks and starts the session. The Connect Status will change from "Listening" to "SERVERCALL", and the small coloured indicator on the bottom right will turn green. jPSKmail uses a 3-way handshake for connect, you know for sure the connect was successful when you receive the 'greeting message' from the server:



```
Connected to 9A1CRA 2013-02-03 18:18

9A1CRA 2.0.12-17:18:59-IM2620>
17 mails.


OK...
```

3. You can now issue any command the server will recognize. The most important functions can be reached with a button or via the menu.
4. To end the session hit the **DISCONNECT** button (same as Connect).
5. If the server does not receive this command, you can abort the session by hitting the **Abort** button. It is courteous to wait for the QUIT to disconnect, in that case the server will not send polls until it times out...

With the **Abort** button you can close a connection one-sided. The client sends an abort frame to the server. If the server does not notice that, it will continue polling until it times out. It is not regarded very courteous to do this.

With the **Pos.** button you can manually send an APRS Posit packet. Any server receiving it will send a QSL packet. You do not have to be connected for this. You can do it also with **<CTRL-B>**.

Terminal tab, chat mode (ARQ)

To use the chat (client to client) mode, just put the callsign of the station you want to call into the server selector and hit the **Connect** button. Automatic mode control is implemented, you can set the initial connect modes for both clients.

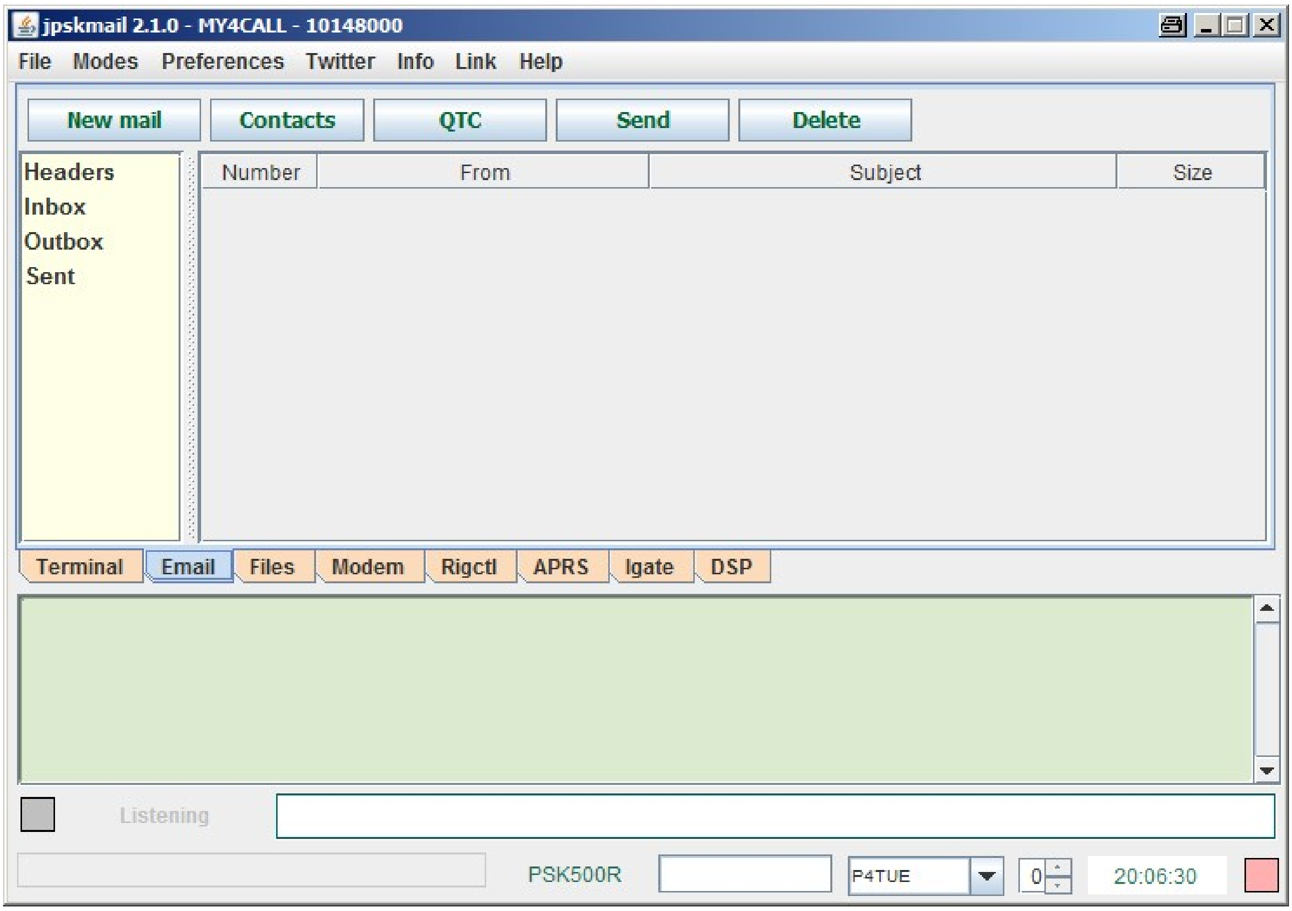
CQ tab

This button will simply send a CQ block. If you get an answer you might send text you enter in the **Textfield**.

Send tab

<TBD>

Email tab



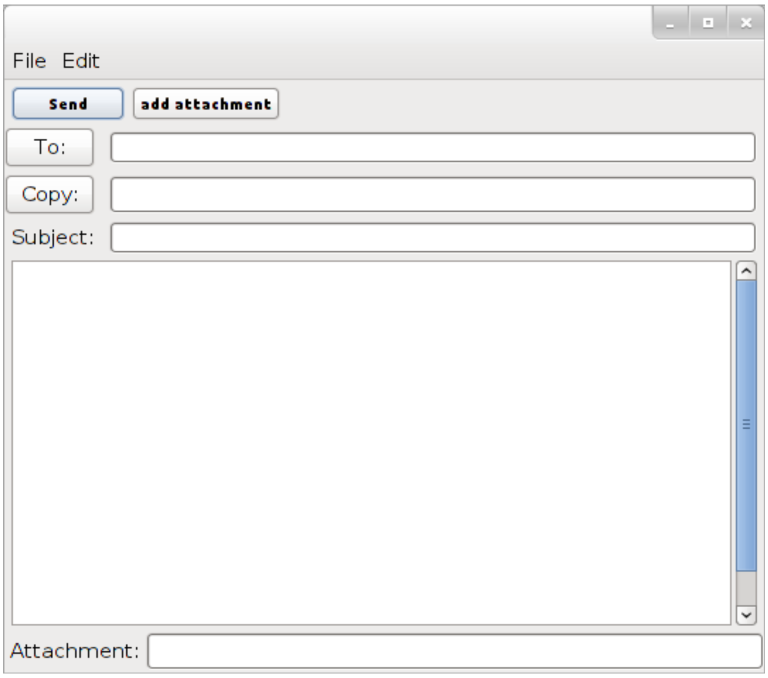
Composing an email

There are two basic ways to create and transmit emails from jPSKmail:

- Proper emails with custom subject and possibly an attachment
- Very short emails, called **pskaprs** emails, with fixed subject and no attachment. See page 37.

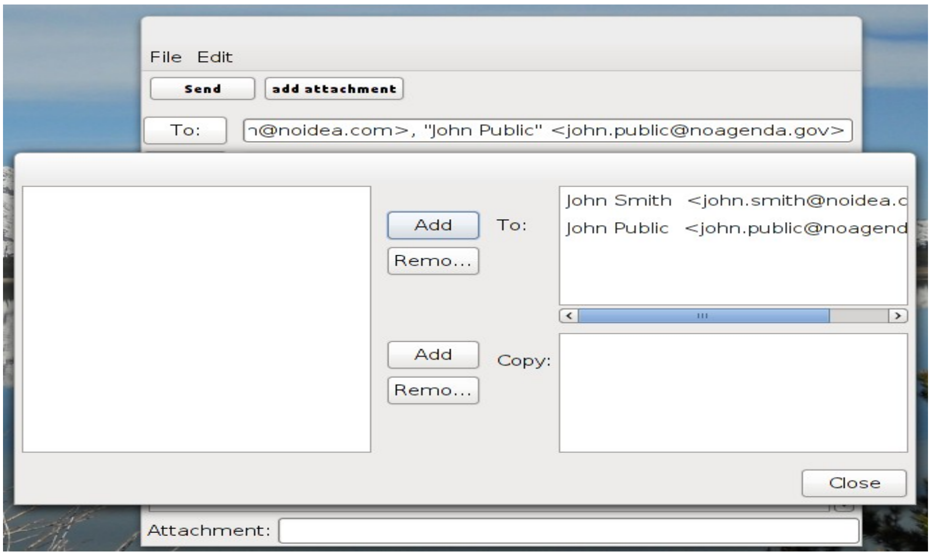
Preparing an email

The email editor can be reached with the **New mail** button. The email editor is rather simple, there are fields for address, copy, subject and message content.



•To
Enter the addressee email address here. Multiple recipients can be entered, its important to then separate them with a comma like: “user1@domain.com, user2@domain.com”

By clicking on the **To** button a window showing the address book contacts is shown and those can be used to address he message. Just select the desired contact on the left hand side and push add for the **To** or **Copy** fields. When done just push the close button and the addresses will be displayed on the new email.



To: ı@noidea.com>, "John Public" <john.public@noagenda.gov>

John Smith <john.smith@noidea.c
John Public <john.public@noagend

•**Copy**
Please note that this functionality is not yet included in the server. In time it will be used to add copy addressees to your message, much like any other email client is able to.

•**Subject**
Well, its the message subject, keeping it somewhat short is good

•**Message body**
The message text, the actual content of the message

•**Add attachment**
If you wish to add an attachment then browse for it after clicking this button. The attachment will be compressed and packed into the message so please bear in mind that it needs to be small.
I would not add attachments bigger than a few kb.

Sending email, connected

•When connected the **Send** button on the email tab can be used to transmit your prepared messages in the local outbox. The **Send** button currently sends one email at a time so if you have two emails then you will have to push that button once, make sure the first email is acknowledged by the server (and removed from the Outbox) and then repeat the procedure.

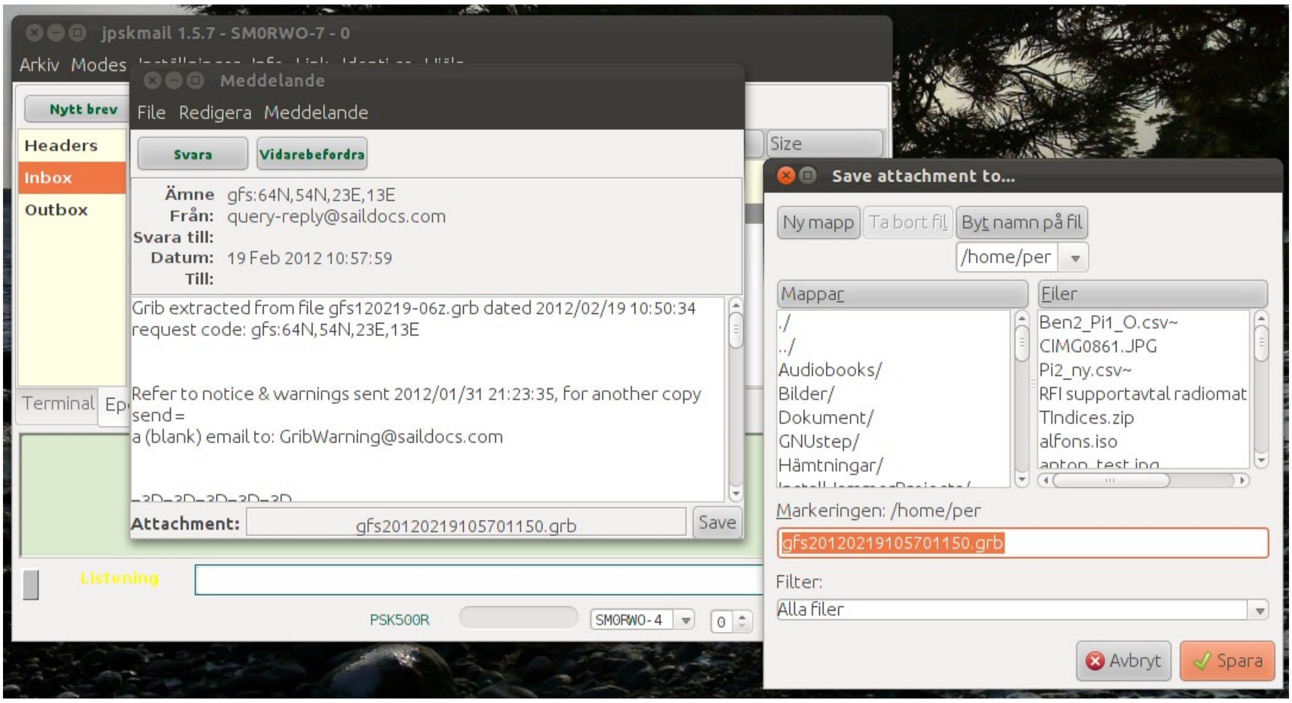
•If you wish to send a certain message in the Outbox first, for instance when sending a message addressed to a local user at the server you are currently connected to, then select that message row in the Outbox before you push send. This will send the selected message first and bypass the message queue. This procedure currently only works when doing compressed email traffic.

Receiving email

Received emails are added to a file called **Inbox**. This file is in mbox format, so you can use it as a spool file for mail clients like Sylpheed or Thunderbird. That way the mail you receive via jPSKmail can be integrated with your regular mail.

To read your mail using jPSKmail,

- **Connect** to the server
- Go to the **Email** tab
- Hit the **QTC** button, your headers will be added to the list.
- When you want to download a mail, **double click on the header**,it will be added to the **Inbox**. Right clicking on the header will give you a short menu (Get or open).
- When you want to read a mail, go to the **Inbox** tab and double click on the header. Again right-clicking will give you a small menu.



Ämne gfs:64N,54N,23E,13E
Från: query-reply@saildocs.com
Svara till:
Datum: 19 Feb 2012 10:57:59
Till:

```
Grib extracted from file gfs120219-06z.grb dated 2012/02/19 10:50:34
request code: gfs:64N,54N,23E,13E


Refer to notice & warnings sent 2012/01/31 21:23:35, for another copy
send=
a (blank) email to: GribWarning@saildocs.com


=3D=3D=3D=3D=3D
```

Attachment: gfs20120219105701150.grb

PSK500R SM0RWO-4 0

/home/per

Mappar: ./ ../ Audiobooks/ Bilder/ Dokument/ GNUstep/ Hämtningar/

Filer: Ben2_Pi1_O.csv~ CIMG0861.JPG Pi2_ny.csv~ RFI supportavtal radiomat TIndices.zip alfons.iso

Markeringen: /home/per

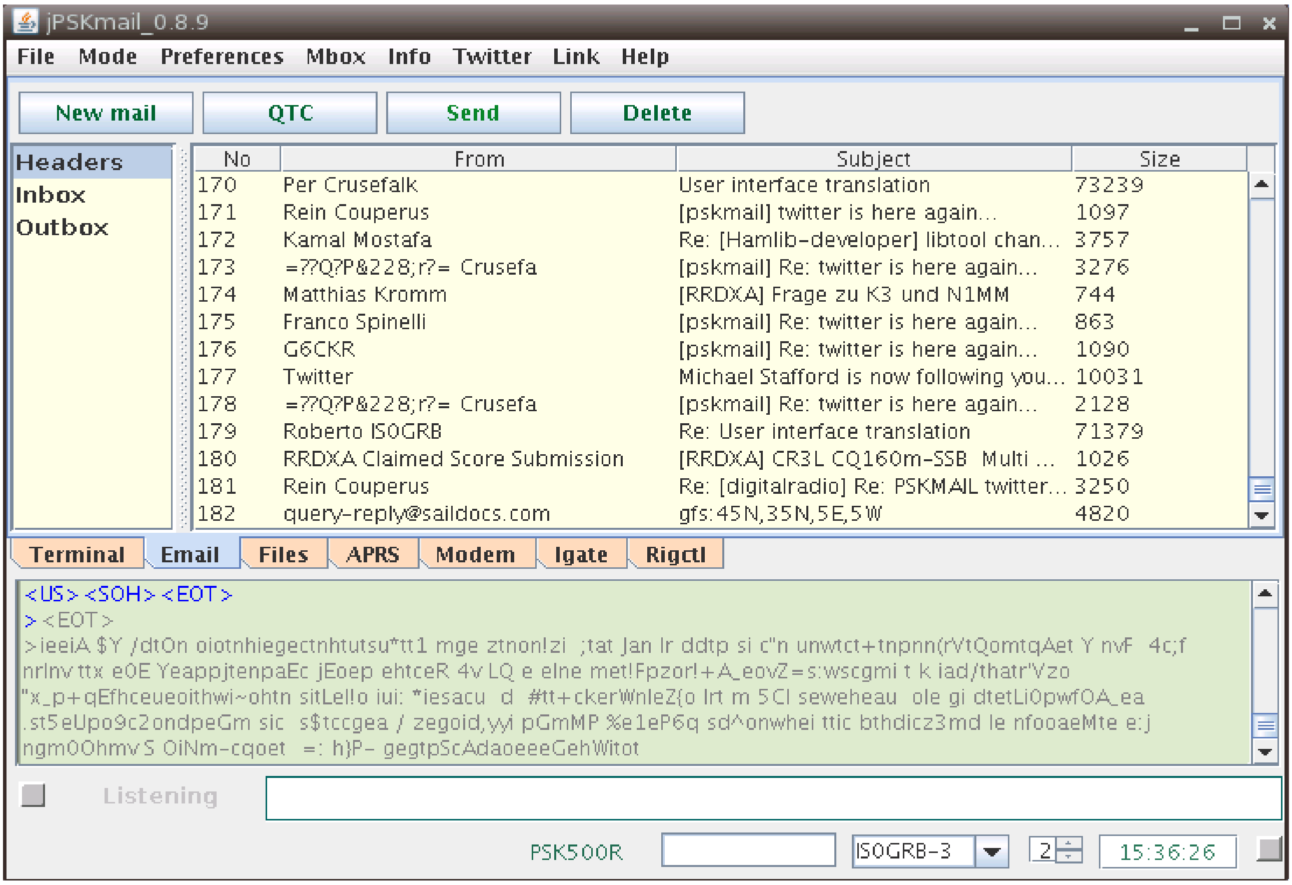
gfs20120219105701150.grb

Filter: Alla filer

Incoming email with attachment (shown at the bottom of the window). Click the save button to save the attachment as a local file.

Mail headers tab

This tab will display the message list fetched from your email account while connected to a pskmail server. You can choose a mail for download by right-clicking on the header. One mail at the time.



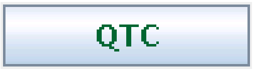
No	From	Subject	Size
170	Per Crusefalk	User interface translation	73239
171	Rein Couperus	[pskmail] twitter is here again...	1097
172	Kamal Mostafa	Re: [Hamlib-developer] libtool chan...	3757
173	=?Q?P&228;r?= Crusefa	[pskmail] Re: twitter is here again...	3276
174	Matthias Kromm	[RRDXA] Frage zu K3 und N1MM	744
175	Franco Spinelli	[pskmail] Re: twitter is here again...	863
176	G6CKR	[pskmail] Re: twitter is here again...	1090
177	Twitter	Michael Stafford is now following you...	10031
178	=?Q?P&228;r?= Crusefa	[pskmail] Re: twitter is here again...	2128
179	Roberto IS0GRB	Re: User interface translation	71379
180	RRDXA Claimed Score Submission	[RRDXA] CR3L CQ160m-SSB Multi ...	1026
181	Rein Couperus	Re: [digitalradio] Re: PSKMAIL twitter...	3250
182	query-reply@saildocs.com	gfs:45N,35N,5E,5W	4820

```
<US><SOH><EOT>
><EOT>
>ieeiA $Y /dtOn oiotnhiegectnhtutsu*tt1 mge ztnon!zi  ;tat Jan Ir ddtp si c"n unwtct+tnpnn(rVtQomtqAet Y nvF  4c;f
nrlnv ttx e0E YeappjtenpaEc jEoep ehtceR 4v LQ e elne met!Fpzor!+A_eovZ=s:wscgmi t k iad/thatr'Vzo
"x_p+qEfhceueoithwi~ohtn sitLel!o iui: *iesacu  d  #tt+ckerWnleZ{o Irt m 5Cl seweheau  ole gi dtetLi0pwfOA_ea
.st5eUpo9c2ondpeGm sic  s$tccgea / zegoid,yyi pGmMP %e1eP6q sd^onwhei ttic bthdicz3md le nfooaeMte e:j
ngm0OhmvS OiNm-cqoet  =: h}P- gegtpScAdaoeeeGehWitot
```

PSK500R ISOGRB-3 2 15:36:26

Mail tab buttons

The **New mail** button starts a dialog which allows composition of a new email. It is possible to add a small attachment (~6k)



The **QTC** button requests the new mail headers from your email account. It first reads the headers file to see what the last header is that was downloaded. It will request only the new headers.
You can override that behaviour by either cleaning the headers file (File->Clear->headers), or send '~QTC +0' To the server. Mail download is done by right-clicking on the header...As soon as the download starts the progress bar will start indicating:

The progress bar goes to zero as soon as it reaches 100%. In Monitor or Listening modes the progress bar shows received signal quality...

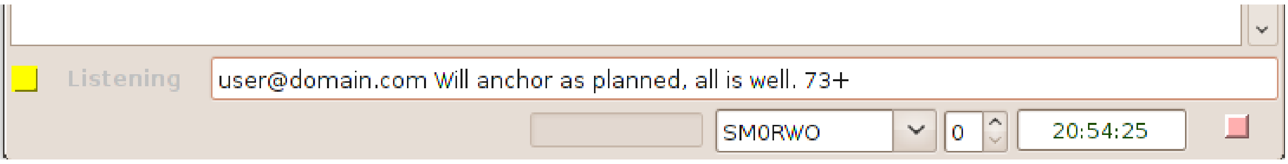
The **Send** button starts the upload of the first email in the Outbox. As soon as **message sent...** has been received from the server the email will be deleted from the Outbox. It is still available in the **Sent** file.

With the **Delete** button you can delete mails from your mailbox at the ISP. Enter the numbers of the mails to be deleted in the **Textfield**. You can enter more than 1 number, separated by spaces. The server will answer with "Mails x x deleted..." in case of success.

The numbers reflect the numbering in your ISP mailbox. After a delete mail operation you should clear your headers list and rebuild it so the list reflects the numbers in the ISP mailbox.

PSKaprs emails

This is a way to transmit a short text with a fixed subject to one email address. Its done in unconnected mode and a server that receives it will transmit an acknowledge packet.

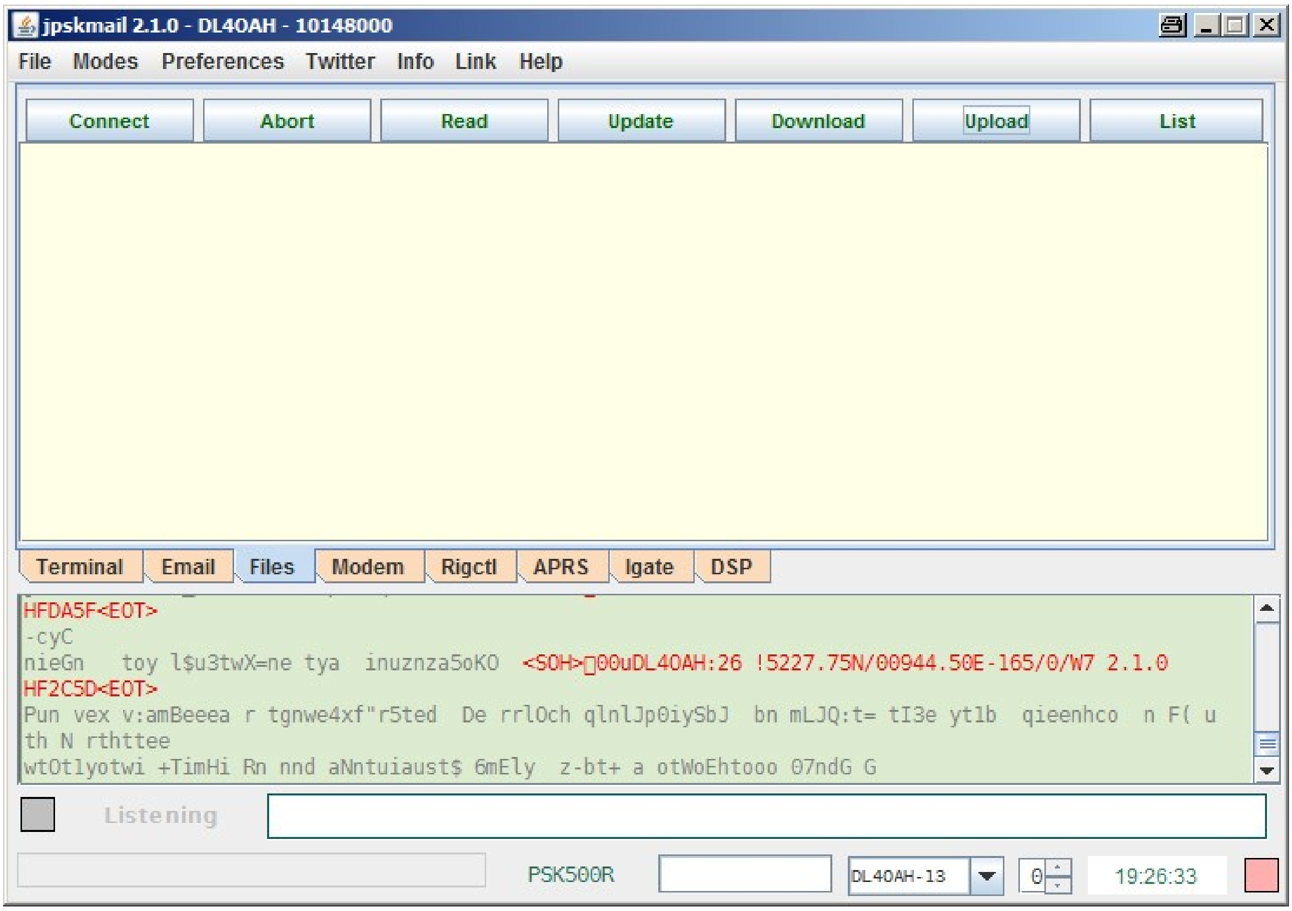


user@domain.com Will anchor as planned, all is well. 73+

This email is sent in unconnected mode so keeping the text short helps it get through to a server.

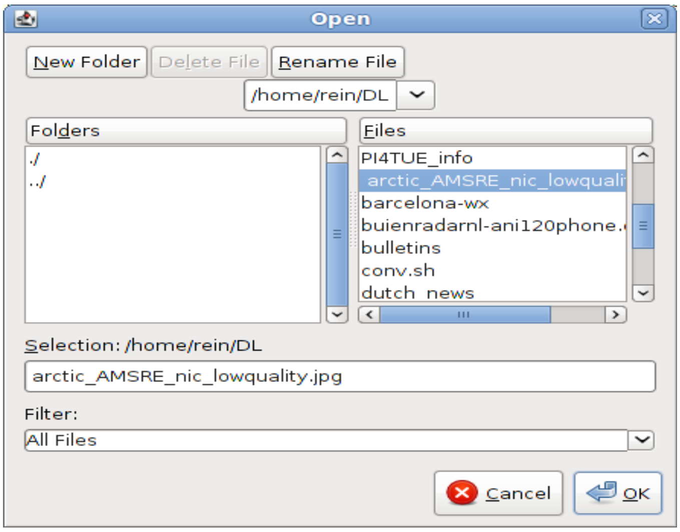
File transfer

For File transfer go to the **Files** tab, and hit the Upload button. After choosing a file to upload, the client will ask you what the addressee for the file is. jPSKmail is being prepared for Delay Tolerant Networking in future.



```
HFDA5F<EOT>
-cyC
nieGn   toy l$u3twX=ne tya  inuznza5oKO  <SOH>▯00uDL4OAH:26 !5227.75N/00944.50E-165/0/W7 2.1.0
HF2C5D<EOT>
Pun vex v:amBeeea r tgnwe4xf"r5ted  De rrlOch qlnlJp0iySbJ  bn mLJQ:t= tI3e ytlb  qieenhco  n F( u
th N rthttee
wtOtlyotwi +TimHi Rn nnd aNntuiaust$ 6mEly  z-bt+ a otWoEhtooo 07ndG G
```

To download a single file from the server, first enter the file name into the input field, or **double click a file name in the 'files list'**, and push the **download** button.File download is binary, compressed with **gzip** and **base64** coded. The client decodes and unzips it and puts it into the Downloads directory.



PI4TUE_info
arctic_AMSRE_nic_lowquali
barcelona-wx
buienradarnl-ani120phone.
bulletins
conv.sh
dutch_news

Selection: /home/rein/DL

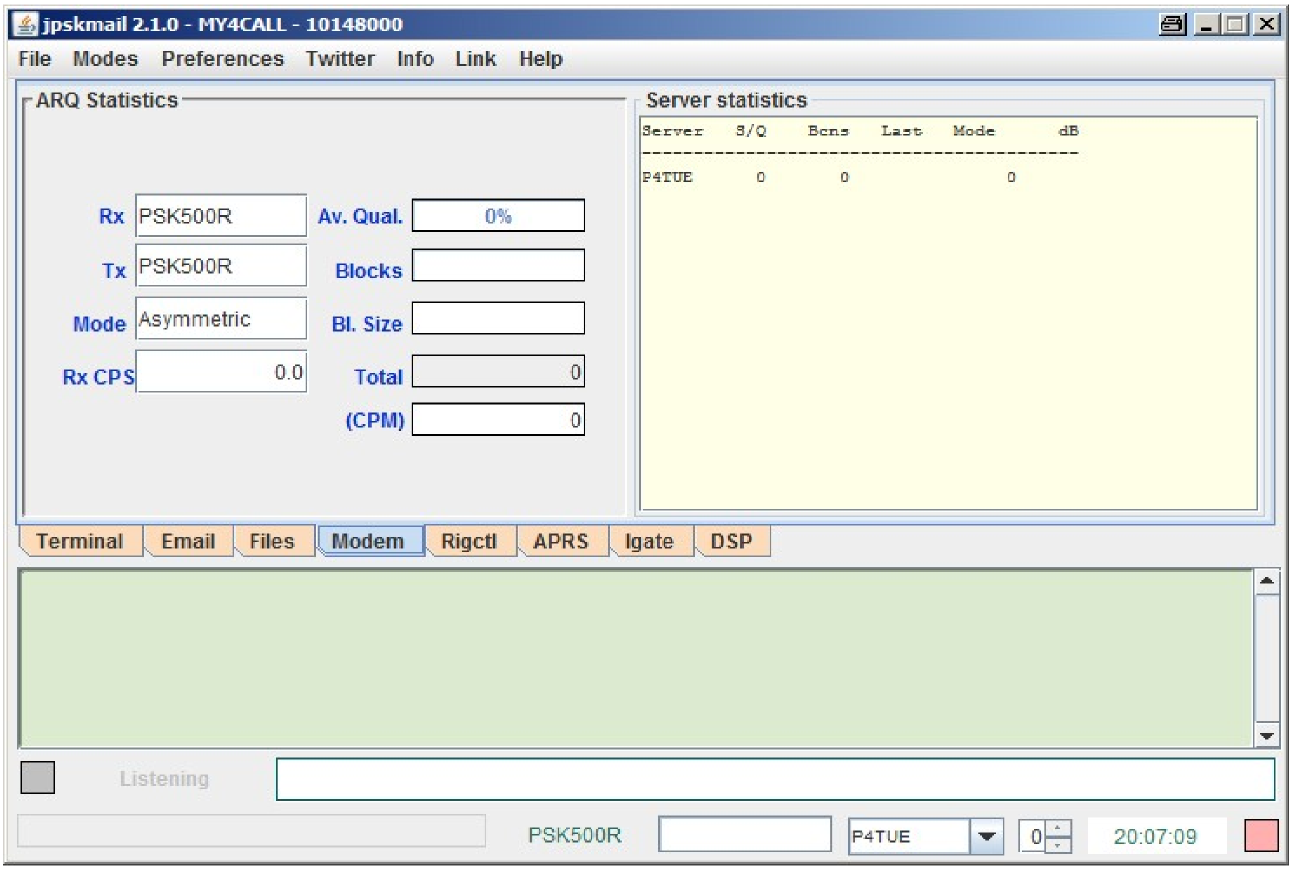
arctic_AMSRE_nic_lowquality.jpg

Filter:

All Files

After download the file can be READ, or UPDATED. There is a file chooser dialog to make that easy.

Modem tab



jpskmail 2.1.0 - MY4CALL - 10148000

ARQ Statistics

Rx PSK500R

Tx PSK500R

Mode Asymmetric

Rx CPS 0.0

Av. Qual. 0%

Blocks

Bl. Size

Total 0

(CPM) 0

Server statistics

```
Server       S/Q    Bcns   Last   Mode        dB
-----------------------------------------------
P4TUE          0       0                0
```

Listening

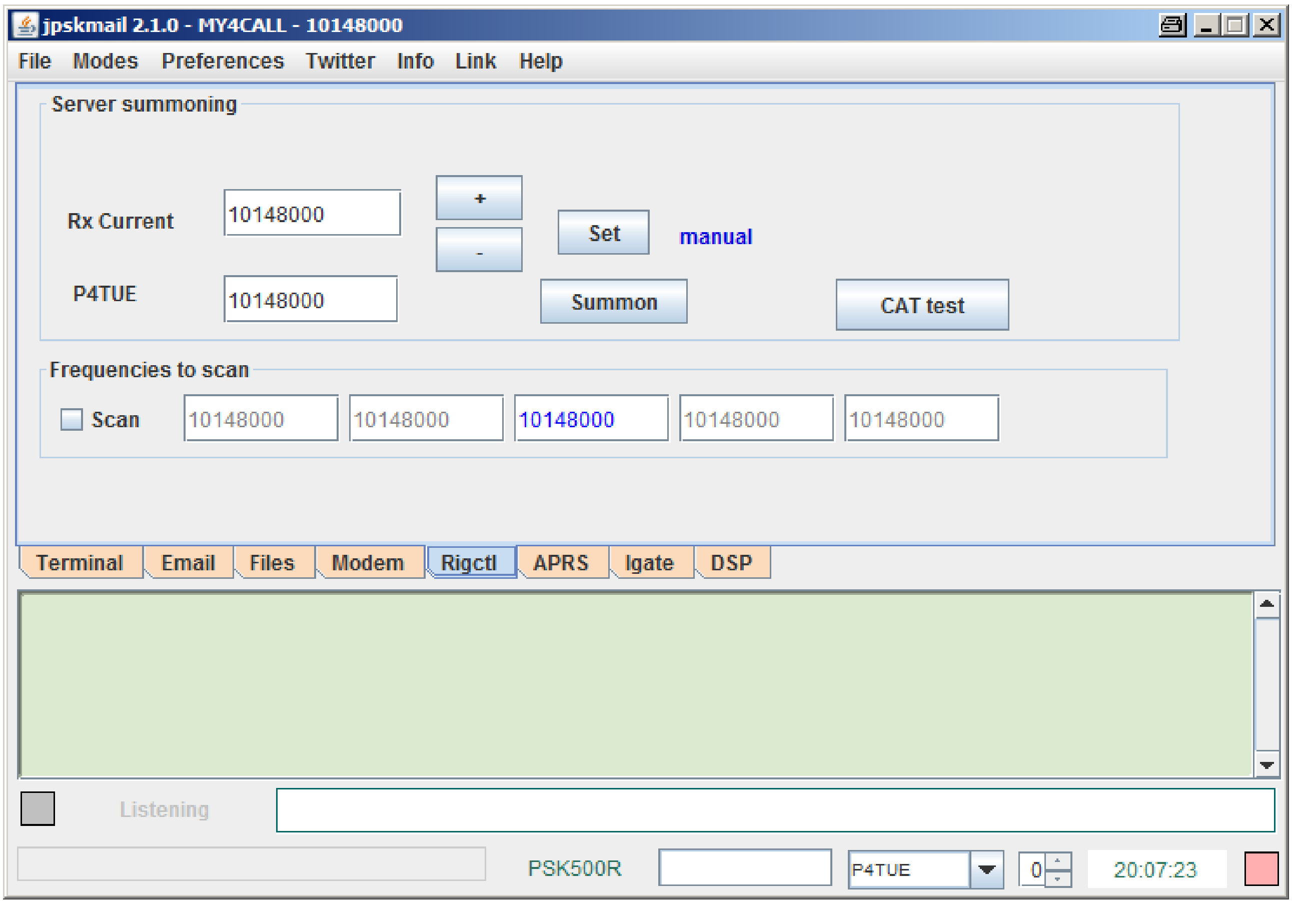
PSK500R P4TUE 0 20:07:09

This tab shows some interesting facts about the current modem status. For instance it shows both the receive and transmit data mode, this can differ during a connected session thanks to the asymmetric linkup capability. S/N-values get fed back immediately and this is can be used to rank heard servers. Rx CPM shows the receive speed (Characters Per Minute).

The maximum possible speed has been measured as 2460 characters/minute (PSK500-PSK500).

There is also an ARQ missing block indicator to show which blocks have to be repeated.

Rig control tab



jpskmail 2.1.0 - MY4CALL - 10148000

Rx Current 10148000

P4TUE 10148000

□ Scan 10148000 10148000 10148000 10148000 10148000

Listening

PSK500R P4TUE 0 20:07:23

The transceiver can be controlled from the **Rigctl** tab when Rigctl is used.
SERVERCALL (in this example **P4TUE**) will follow the server frequency table. The **Set** button will put this into **Rx Current** , which is the CAT frequency of the rig.

Rx Current can be incremented and decremented in 500 Hz steps with the '**+**' and '**-**' buttons. The transceiver will follow.

The **Summon** button is special:

If the **Rx Current** and *SERVERCALL* frequencies are different, the **Summon** button will:

- Issue a **Connect Request** on the ***SERVERCALL*** frequency
- **Listen** for the acknowledge on the **RX Current** frequency.

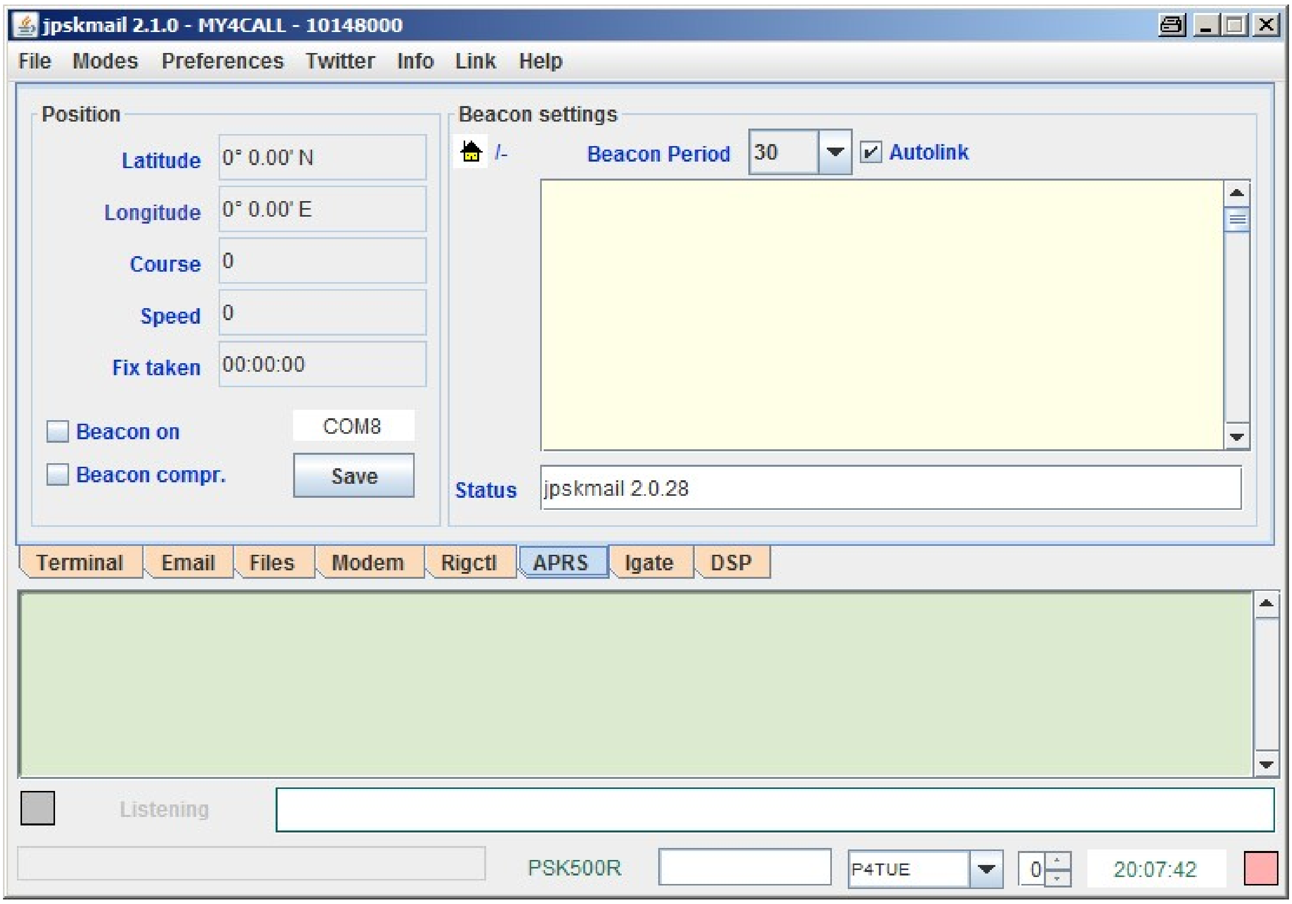
This tells the server to come to your **Rx Current** frequency to handle the traffic.

Of course this only works if the server is configured to support summoning.

The scanning function can be switched on separately. You can also enter the default frequencies for the scanner. Initially the frequencies are taken form the **Rig** tab in **Preferences** settings. Values must be be entered as the center frequencies of the channel, as the scanner will use the Offset to calculate the VFO dial frequency. It is possible to store the frequency tables of known servers in the ~/.pskmail directory. The files are called SERVERCALL.chn, and contain a comma-separated list of frequencies (in Hz).

APRS tab, Unconnected Operation

Unconnected operation (like UI frames in packet radio) is done from the **APRS** screen.



jpskmail 2.1.0 - MY4CALL - 10148000

Position

Latitude: 0° 0.00' N

Longitude: 0° 0.00' E

Course: 0

Speed: 0

Fix taken: 00:00:00

□ Beacon on

□ Beacon compr.

COM8

Beacon settings

/-

Beacon Period: 30

☑ Autolink

Status: jpskmail 2.0.28

Listening

PSK500R

P4TUE

0

20:07:42

From this screen you can organize position beacons, send APRS messages to other pskmail stations, to VHF units, or to the internet (Uiview, Xastir), or send APRS short emails to any email address.

You do this via a PSKmail server which is a gateway to the internet. Servers are interconnected through the APRS Internet backbone (APRS-IS).

It Is common practice to set the APRS beacon period to 30 minutes or 60 minutes for non-moving objects.

To **send a position beacon** manually you can use **<CTRL-B>** or the **Pos** button on the Terminal tab.

To send an APRS text message just enter “**CALLSIGN Texttexttext**” into the **Textfield** at the bottom of the screen. Use UPPER CASE for the callsign. If the callsign is known at a VHF APRS digipeater it will be delivered to VHF.

To send a short email enter “**email@addres.xxx** **Texttexttext**” into the **Textfield**.

- **Latitude**

  Latitude value in degree / decimal minute format from GPS if available or from Preferences / User tab.

- **Longitude**

  Longitude value in degree / decimal minute format from GPS if available or from Preferences / User tab.

- **Course**

  Course value in degree from GPS if available.

- **Speed**

  Speed value in km/h from GPS if available.

- **Fix taken**

  Fix taken value in hh:mm:ss format from GPS if available.

- **Comp.**

  Check this to send compressed beacons. These are shorter, but not human readable. Yoe should use it to save valuable shortwave bandwidth.

- **Beacon on**

  Must be checked to automatically send beacons in the selected Beacon Period interval.

- **Beacon Period**

  Combobox control to select the time in minutes between beacons. Don't use values below 30 minutes except for testing purpose.

- **Status**

  Textfield which is part of the beacon. Like “PA0R on the road” or “SY Bilbo relaxing at anchor after a stormy day”

- **Autolink**

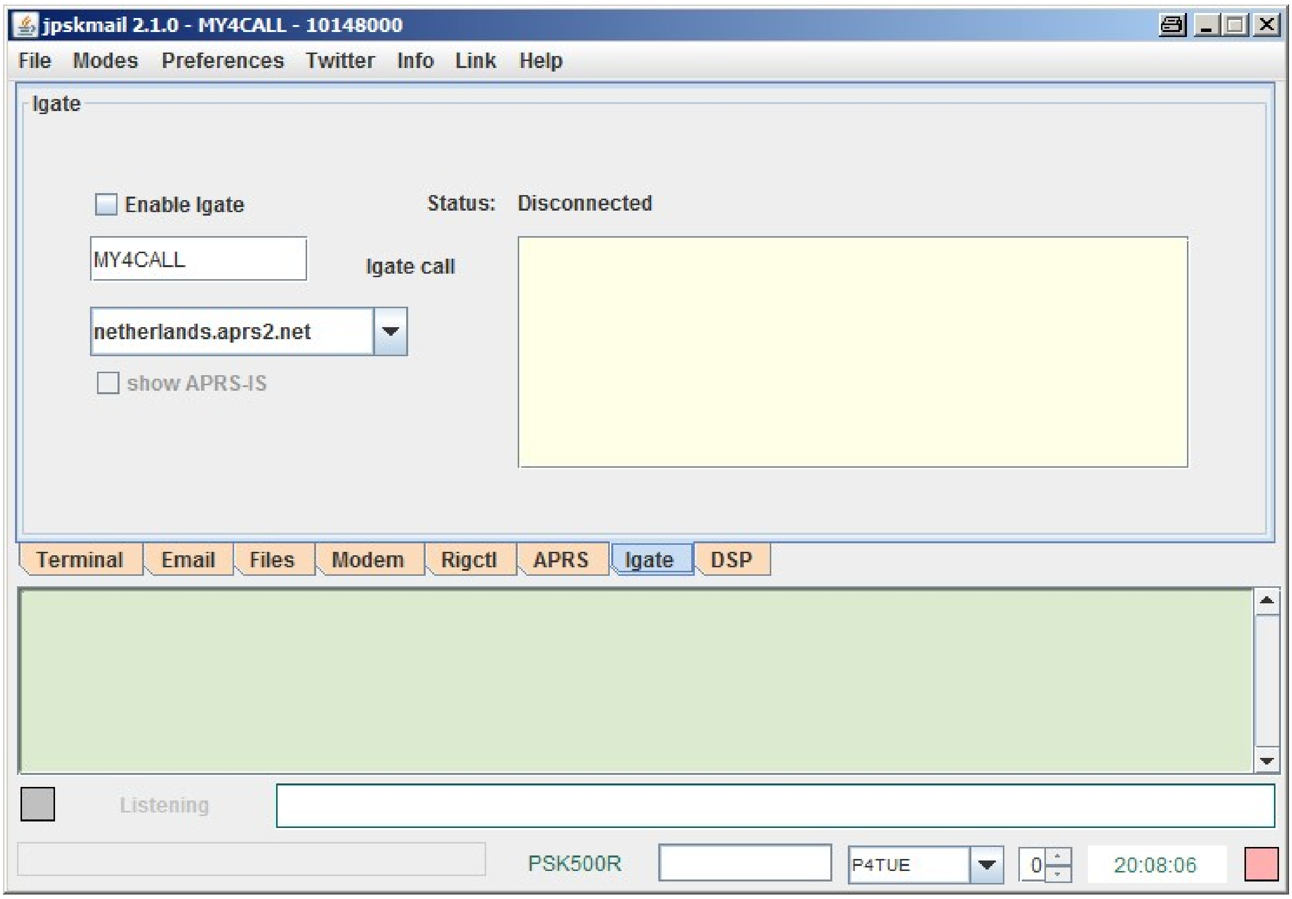
  This checkbox has nothing to do with APRS, but with UI frames. When checked jPSKmail will send a **link block** to the selected server every 10 minutes. You can doe this also via Menu / Link / Link or with **<CTRL-L>**.

- **Save button**

  Click this button if you want to make the settings on this tab permanent. The settings of **Igate** tab are also saved. **<CTRL-S>** also saves these settings and all other.

Igate Tab

The **Igate** can be activated on the **Igate** tab. When it is on, it will connect to a APRS Tier2 server. All beacons and messages received by the **Igate** will be sent to the APRS-IS backbone.



jpskmail 2.1.0 - MY4CALL - 10148000

Igate

□ Enable Igate

Status: Disconnected

Igate call: MY4CALL

netherlands.aprs2.net

□ show APRS-IS

Listening

PSK500R

P4TUE

0

20:08:06

- **Enable Igate**

  Check here to enable the igate function.

- **Igate call**

  Enter the call used for identification at the Tier 2 Server. The password is calculated by the program.

- **Tier 2 server**

  Select the Tier 2 Server you want to use. Normally a server in your region.

- **Status**

  Shows connected or disconnected. The text panel below shows the records sent to the APRS-IS.

To make your settings permanent you must use the **Save** button on the **APRS** tab or **<CTRL-S>**.

General remarks

The link password

Connect to the server. Hit **<CTRL-U>** or go to Preferences->Email data and do a **Update server**. The link password is automatically sent to the server after your mail user record. You have to do this only 1x for every server used (unless the server looses its database). The link password is used to prevent misuage of your callsign via pskmail.

Beacon minute and server channel

A server sysop may want to cover several frequencies, using only one transceiver. By switching between several bands its possible to have both local coverage on 80 meters and DX on higher bands, provided the antenna system is capable of that. jPSKmail uses a five minute continuous scan loop where servers can monitor one frequency per minute. The beacon minute setting in the client controls what minute to send APRS position beacons, a correct value there is then the minute that the server you want to link to listens to the frequency you are tuned to. The server scans during minute 0-4 and can use one default mode and frequency per minute. The scan loop goes on and on so minute 5 of the hour is minute 0 in the loop and so on.

Let's say that the server scans 80,40,30,20 and 17 meters. It listens during one minute on each frequency and all the servers update their scan table on the wiki, that way you can find what frequency and mode the server is on. So, if you would like to try 30 meters then the server in the example is there during minute 2. So, you should set minute 2 on the client and it will handle beacons etc. on the proper minute.

This means its important to sync your pc clock using ntp or manually. There are numerous ways about this, from GPS (GPS time is visible beneath the position on the APRS tab if you are connected to a GPS). At this time jPSKmail will not update the system time, the reason for that is that it requires different user privileges on different operating systems.

During a connected session, while the Monitor function is active, and also during reception of a Bulletin, scanning is halted and the client will remain on the channel..

Choosing the appropriate modes

Receiving conditions can be completely different at server and client. Often the server runs more power than the client, and the client's antenna is often too small, badly tuned and in the middle of an electro-magnetic field we usually call 'local QRM'. The latter is typical for camp sites with lots of RV's, or boats in a harbour. That is why PSKmail can use **asymmetric link control,** providing different modes for uplink and downlink. Link control for both up- and downlink is done by the server. Mode changing is established using RSID, a separate super robust MFSK signal telling the modem at the other end of the link which mode to listen in. Proper reception of RSID signalling is critical to efficient ARQ operation, and it is the lower limit of what can be achieved on a link. It is important to choose modes which are capable of connecting to the other end of the link. A good choice for the client mode is the default listening mode of the server (PSK500R or PSK250), THOR22. or MFSK16. Once the connect has been established, the server will take over mode control, and will use the proper modes for up- and downlink channels. If you are not sure your frequency is exact, you can try THOR22, which is largely insensitive to small frequency offsets. The new Connect window allows you to set different RX and TX modes for the connect sequence.

Please be aware that slow modes need more time to get a connect packet through to a server. It pays to start connecting at the beginning of the minute. The newly introduced

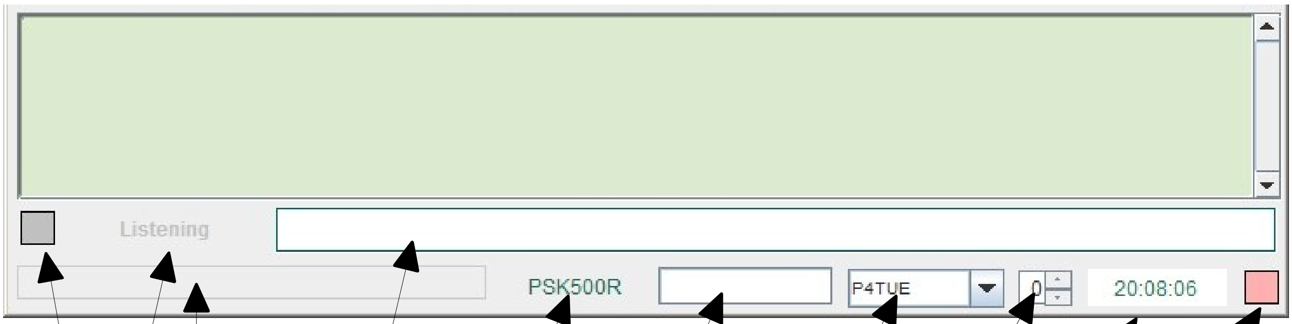
THOR4 mode will even need 90 seconds, so it can only be used for connecting if the server actually listens in THOR4 for minimum 2 minutes!!

Bulletin reception mode

Whenever the client receives a bulletin (anything starting with '**ZCZC**' and ending with '**NNNN**'), it will go into bulletin mode. Several things happen in Bulletin mode:

- Frequency scanning is stopped during reception of the bulletin
- The received text will be appended to the file ~/.pskmail/Downloads/bulletins.

Bottom part of the client



The **status line** shows what the client is doing, **Listening** most of the time.

When you click the Status Line it will switch to **Monitor mode**. In Monitor mode incoming packets are shown with a time stamp, and if available the session partners...

Monitor mode stops automatically as soon as a connected session is established, or when a **bulletin** is received. Also, **Monitor mode halts the frequency scanning,** so you are effectively monitoring a **single channel**...

Status Indicators

There are 2 status indicators, the channel indicator and the link status indicator. The left hand one is the **channel indicator**. The colours mean:

- Grey Idle, no data
- Yellow Data, not valid
- Blue Data block in progress
- Red Transmitting data

There is a slight delay, because the client does not know the link status until it receives it from the server. i.e. most of the time the link indicator is 1 frame behind.

Temporary **program status** messages.

The **Textinput** field is used for parameters, commands etc. as described in this manual.

Current **Rx mode**, **progress** bar for downloads etc.

Selection control for the s**erverlist** of preset or heard servers. Editable: you may enter a temporarily desired server call.

Spinner control for **beacon minute** to allow temporarily change at runtime.

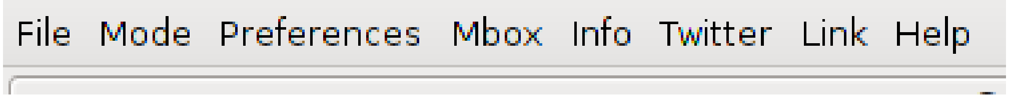
Dislplay of the current **system time**.

The right hand indicator shows **link status.** The colours mean:

- Green No errors on the link
- Yellow RX error
- Blue TX error
- Red RX + TX errors

The large text area shows the protocol to the sound system. Outgoing data are always in red, incoming data different colours due to their type.

Main menu



The main menu is divided into these main sections:

- **File**

  General operations like connecting to a server or client, clearing lists and closing the app.

- **Mode**

  Controls the main mode of the client and the modem mode including mode profiles for adaptive speed during connected sessions

- **Preferences**

  The way to edit or save the preferences dialog

- **Mbox**
- The way to handle local messages on a server during connected session. This menu item is only visible if there is local mail waiting at the server you are linked to.
- **Info**

  Commands to send to a server during a connected session
  See a detailed breakdown of Info further below.

- **Twitter**

  Send and get status updates on Twitter

- **Link**

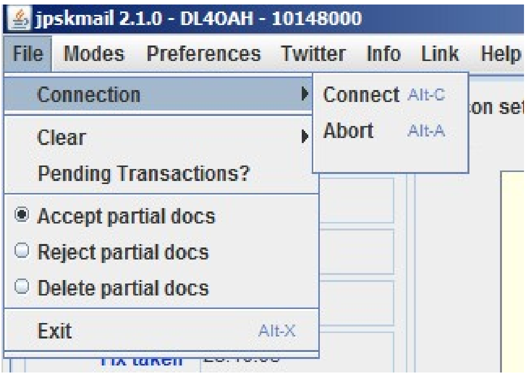
  Here is where we handle the link to servers

- **Help**

  Opens the interactive help which will be created from this manual and displays license information about the application.

File

Handles general operations like connecting, clearing lists and closing the application.



- •Connection
- •Clear Inbox, or Outbox
- •Clear headers
- •Clear bulletins
- •Show pending transactions
- •Set file up/download resume policy
- •Exit **<ALT-X>**

Mode

Controls the main mode of the client and the modem mode including mode profiles for adaptive speed during connected sessions. For a description see Appendix A

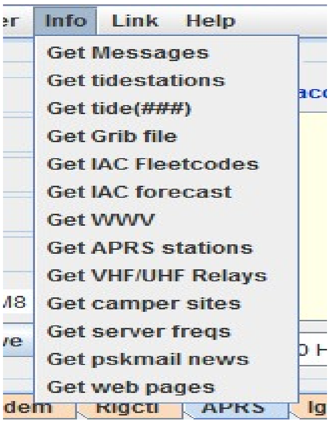
Preferences

- •Edit: open the preferences dialog **<CTRL-E>**
- •Save; save all current setting **<CTRL-S>**
- •Frequency calc: helper window to calculate dial- and center frequency
- •Floating WF: display the waterfall on a small separate window

Mbox

The way to handle local messages on a server during a connected session. By local messages we mean messages addressed to *user@server*. These messages are stored locally , or forwarded to other PSKmail servers. The Mbox menu is only available when there are messages pending at the server you are linked to. You can send mails addressed to a server by using the *call@servercall* mail address.

Info



- **Get Messages**

  Fetch APRS messages for user stored at findu.com

- **Get tidestations**

  Get list of tidestations close to current position

- **Get tide(###)**

  Get tide data for tide station ###, use get tidestations to get number of the wanted tide station.

- **Get Grib file**

  Here the client will send an email to *query@saildocs.com* requesting grib data, centered on your position +- 5 degrees. The server needs to know your return email address. Send your mail record to the server if that has not been done. The grib file is delivered to your mailbox in the cloud. Download it and view the file with the **zyGrib** program.

- **Get IAC Fleetcodes**

  This will query NOAA and fetch the IAC fleetcodes that produce a current pressure map for the north Atlantic. A file will be downloaded to the client and placed within ~/.pskmail/Downloads. You can view the file with zyGrib.

- **Get IAC forecast**

  This will query NOAA and fetch the IAC fleetcodes that produce a 24 hour forecast pressure map for the north Atlantic. A file will be downloaded to the client and placed within ~/.pskmail/Downloads. zyGrib is a good choice for viewing these files.

- **Get WWV**

  Get latest Geophysical Alert Message wwv.txt from NOAA.

- **Get APRS stations**

  Get a list of nearby APRS stations, uses clients reported position. When Xastir or Uiview is connected the APRS stations will appear on the map.

- **Get VHF/UHF Relays**

  Get VHF/UHF Relays from DARC.de centered at latest reported position.

- **Get camper sites**

  Get nearby camp sites from DARC.de. Works only in EU.

- **Get Server freqs**

  Get list of server frequencies entered at the PSKmail wiki (server page).

- **Get pskmail news**

  Get the latest pskmail news from the PSKmail wiki (may be outdated).

- **Get web pages**

  Open the web page window where a web page can be requested from the server.

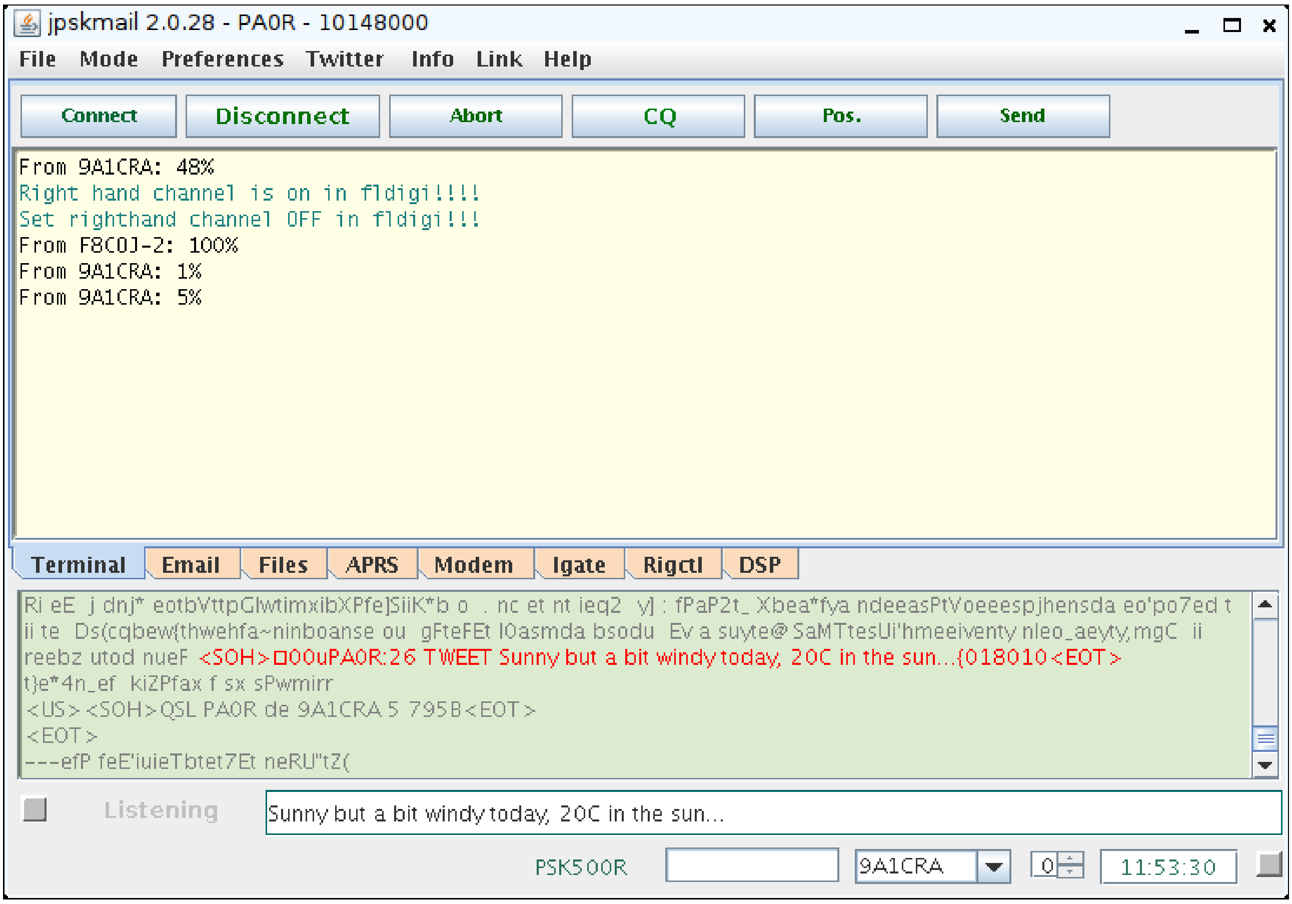
Twitter

Send and get status updates for the PSKmail channel on Twitter.

You can use this service by opening a free account an Tweetymail.com under your CALL.

E.g. my account is PA0R@Tweetymail.com.

Just write your message and send it via the Twitter menu or use **<ALT-S>**. The jPSKmail client sends this to the server:

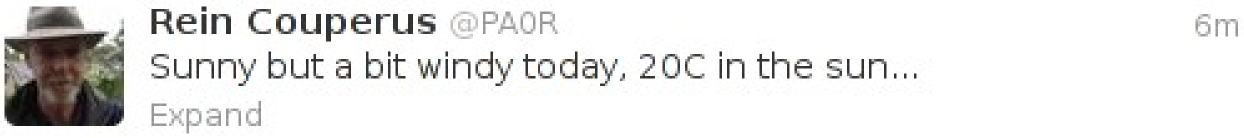


```
From 9A1CRA: 48%
Right hand channel is on in fldigi!!!!
Set righthand channel OFF in fldigi!!!
From F8COJ-2: 100%
From 9A1CRA: 1%
From 9A1CRA: 5%
```

```
Ri eE  j dnj* eotbVttpGlwtimxibXPfe]SiiK*b o  . nc et nt ieq2  y] : fPaP2t_ Xbea*fya ndeeasPtVoeeespjhensda eo'po7ed t
ii te  Ds(cqbew{thwehfa~ninboanse ou  gFteFEt l0asmda bsodu  Ev a suyte@ SaMTtesUi'hmeeiventy nleo_aeyty,mgC  ii
reebz utod nueF <SOH>□00uPA0R:26 TWEET Sunny but a bit windy today, 20C in the sun...{018010<EOT>
t}e*4n_ef  kiZPfax f sx sPwmirr
<US><SOH>QSL PA0R de 9A1CRA 5 795B<EOT>
<EOT>
---efP feE'iuieTbtet7Et neRU"tZ(
```

Sunny but a bit windy today, 20C in the sun...

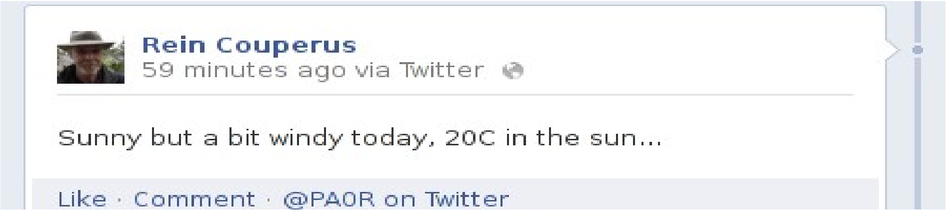
The server receiving this (in this case 9A1CRA) sends the text to PA0R@tweetymail.com, and 1 second later it appears on twitter:



Rein Couperus @PA0R 6m

Sunny but a bit windy today, 20C in the sun...

And if you have coupled twitter with facebook, it will also appear as a facebook update:



Rein Couperus

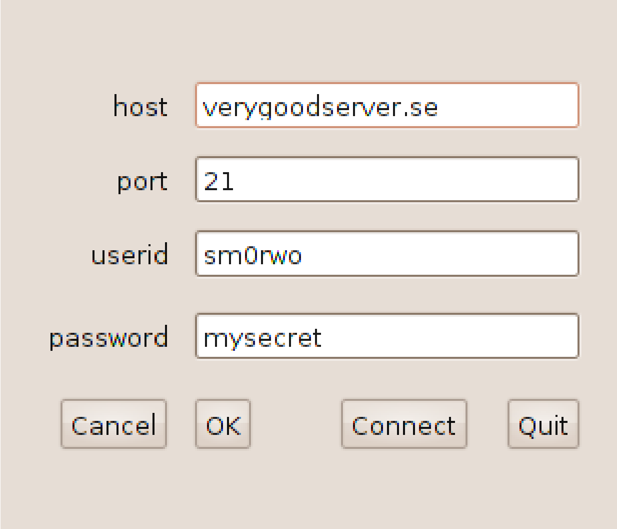
59 minutes ago via Twitter

Sunny but a bit windy today, 20C in the sun...

Link

Here is where to handle the link to servers:

• Ping	Send a ping frame on frequency	**<CTRL-P>**
• Inquire	Send a ping to one specific server	**<CTRL-I>**
• Quality	Show signal quality table on main terminal	**<CTRL-Q>**
• Link	Send a link request to the server	**<CTRL-L>**
• Update	Update server with your mail data record	**<CTRL-U>**
• Telnet	Start the telnet agent	**<CTRL-T>**



host verygoodserver.se

port 21

userid sm0rwo

password mysecret

Illustration 1: Telnet connection dialogue

Help

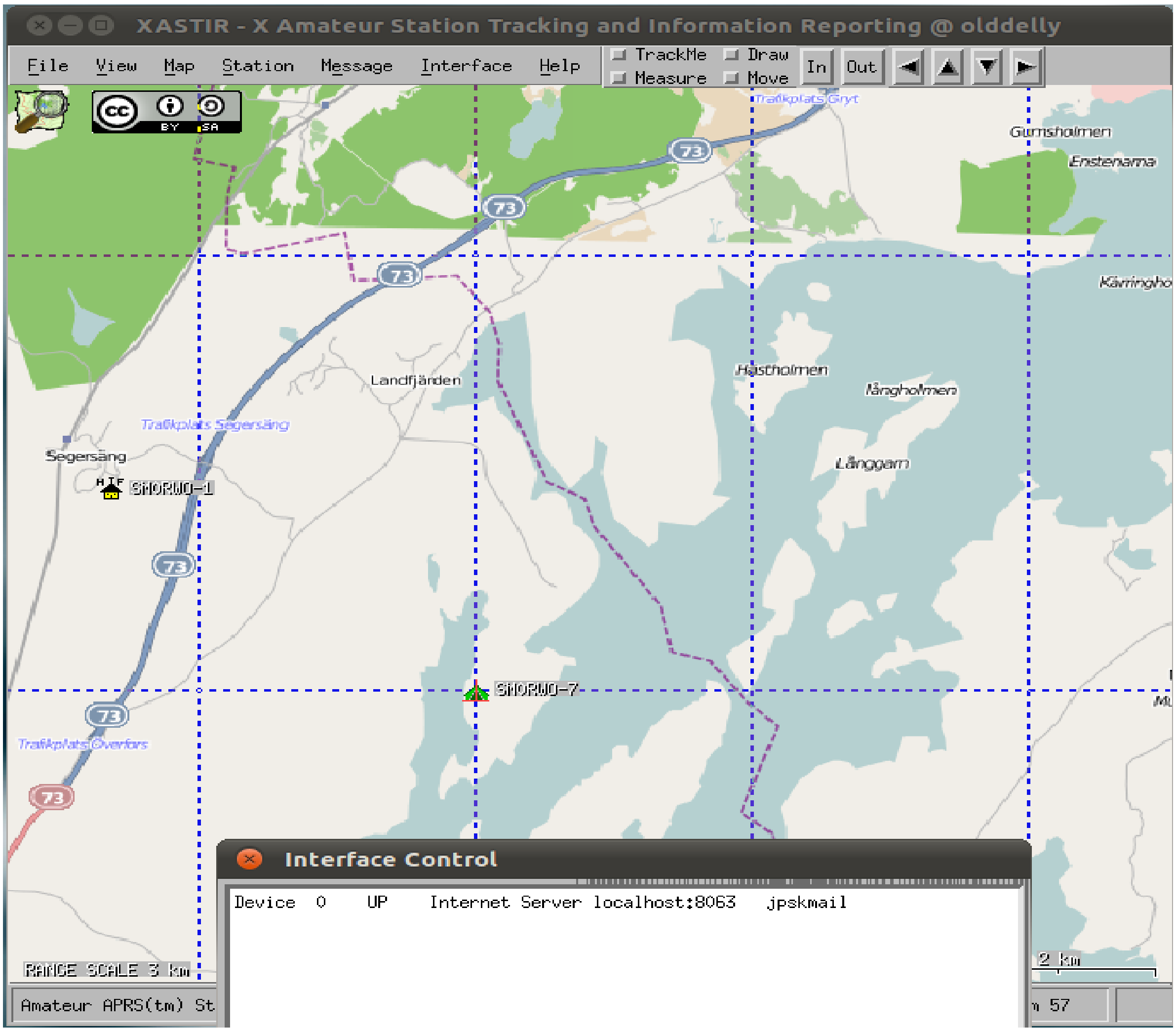
Opens the interactive help which will be created from this manual and also displays license information about the application.

Mapping application

APRS information is mostly geographical by nature. Most often we transfer our position and while doing so others can and will receive that data. During a connected session we may ask about stations in the vicinity and the server will return a long list of stations with position attached.

The geographical data can be very useful as text but even more so when displayed on top of a nice map. JPSKmail includes a server socket that many APRS clients can be very easily connected to. The APRS server is enabled within options, please have a look at the chapter here that deals with the devices tab. Just set that to enabled and select a port to use (or use the default 8063 which should be fine in most cases).

Beneath is an image where xastir, http://www.xastir.org/, is connected to jPSKmail running on the same computer. The procedure is simple, just ask xastir to connect to an internet server located on localhost at the port set within options (default is 8063).



```
Device  0    UP    Internet Server localhost:8063    jpskmail
```

How to get weather information through PSKmail

This section is mostly intended for sailors but can of course be useful also for other purposes and situations. I am mostly writing as a sailor as that is the situation familiar to me.

There are multiple ways to get weather information, there are also multiple ways of using this information and the intention here is to briefly talk about all of this. In no way is this the complete resource on how to do this as that would require several books or years of study.

The three main ways to get weather data at the moment is as clear human readable text (i.e. the weather report or forecast), IAC Fleet codes and grib files. The latter two are intended for use with a viewing tool.

There are also other possibilities, weather fax stations are available on the HF bands and a server sysop can set up a job that makes sure the latest report, forecast or image is available at the server all the time (contact the system operator for special needs).

Weather reports, the human readable kind

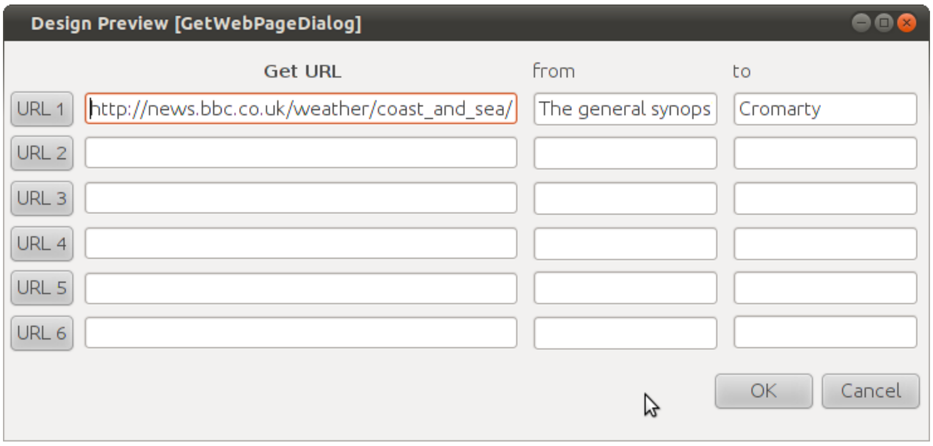
The main ways of getting these reports and forecasts are three:

1. Get a web page containing the necessary weather data
2. Download a file from the servers file area
3. Listen to a broadcast from a server

Fetch a web page

Getting a weather forecast from a web page involves a few steps. The first step is finding a reliable source web page to get data from. There are of course numerous sources for these and the PSKmail wiki is a good place to list the ones that work now. Here I will use BBC as an example, for instance the url: http://news.bbc.co.uk/weather/coast_and_sea/shipping_forecast/#area-4

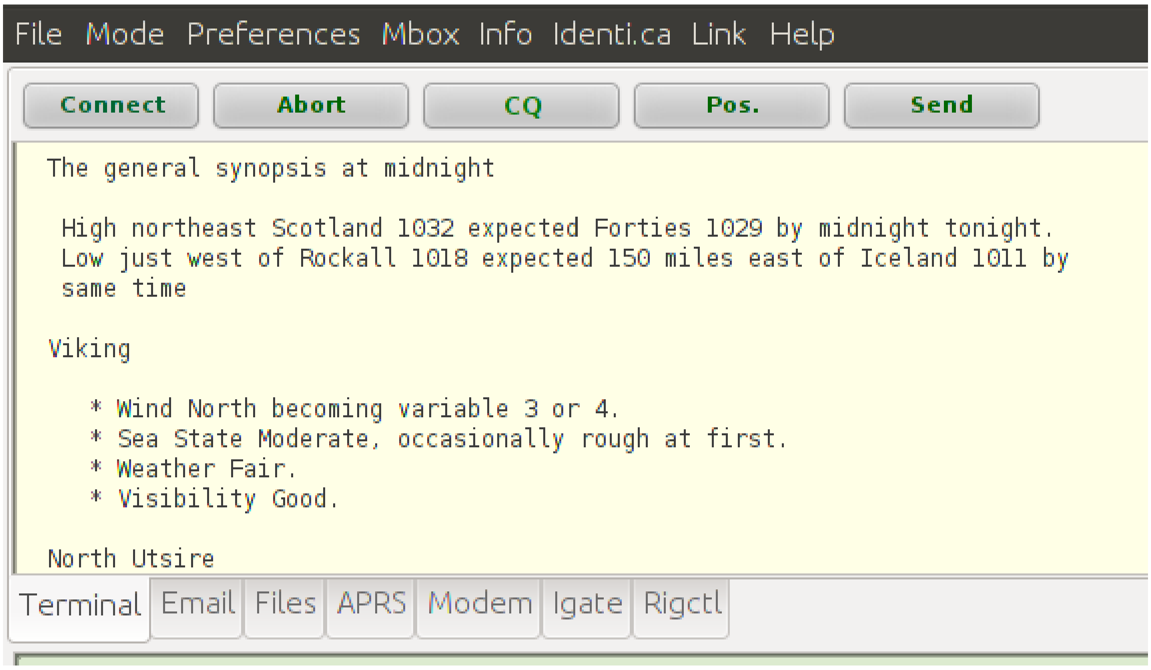
As PSKmail is not a broadband connection to the internet its a good idea to limit the amount of data requested and just get the necessary text. By browsing this web page its easy to see that we can cut away things before and after the area we are looking for, for instance we can start getting the page from “general synopsis” and end at “Cromarty”. This way we can get just the part we are interested in. The web page request window can be prepared like the image below.



Design Preview [GetWebPageDialog]

	Get URL	from	to
URL 1	http://news.bbc.co.uk/weather/coast_and_sea/	The general synops	Cromarty
URL 2			
URL 3			
URL 4			
URL 5			
URL 6			

When the web page bookmark has been prepared we are ready to connect to a server and push that button to request the page. When the server returns the requested page it will look like the example image below:



```
The general synopsis at midnight

 High northeast Scotland 1032 expected Forties 1029 by midnight tonight.
 Low just west of Rockall 1018 expected 150 miles east of Iceland 1011 by
 same time

Viking

    * Wind North becoming variable 3 or 4.
    * Sea State Moderate, occasionally rough at first.
    * Weather Fair.
    * Visibility Good.

North Utsire
```

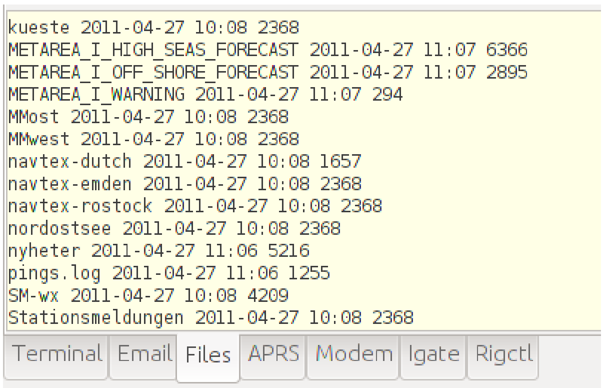
So, in summary what is needed to request a web page fetch of the weather forecast is:

1. Find a web page with a suitable weather forecast (look to the wiki for suggestions)
2. Prepare the web page request bookmark
3. Connect to a server and
4. During the connected session push the web page button and request the weather page.

Also, the email compression selector will affect the way web pages are fetched. If compression is enabled then the web page will be asked for in a zipped format (TGETZIP) and if not in clear text.

Download a file from the servers file area

Servers are connected to the internet and can fetch weather data periodically. This means that forecasts, warnings and even images etc. can be available on the server ready for download to the client. The server file area can be accessed during a connected session with a server. Connect to the server and select the **Files** tab. To see what files are available on the server just click on the List button. The server will respond with the current list of files, like the image below:



```
kueste 2011-04-27 10:08 2368
METAREA_I_HIGH_SEAS_FORECAST 2011-04-27 11:07 6366
METAREA_I_OFF_SHORE_FORECAST 2011-04-27 11:07 2895
METAREA_I_WARNING 2011-04-27 11:07 294
MMost 2011-04-27 10:08 2368
MMwest 2011-04-27 10:08 2368
navtex-dutch 2011-04-27 10:08 1657
navtex-emden 2011-04-27 10:08 2368
navtex-rostock 2011-04-27 10:08 2368
nordostsee 2011-04-27 10:08 2368
nyheter 2011-04-27 11:06 5216
pings.log 2011-04-27 11:06 1255
SM-wx 2011-04-27 10:08 4209
Stationsmeldungen 2011-04-27 10:08 2368
```

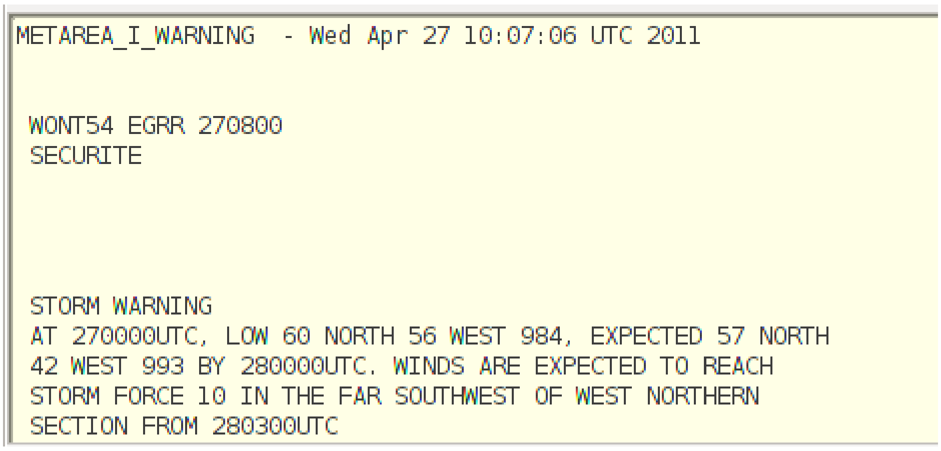
Files are listed with filename, date when it was last fetched and size in bytes.

The size part is important, get the bigger files when the connection is good.

To get one of the files just click on it to select it and then use the Download button located just above the text control. The client will then request the file from the server.

The available files and content will vary with the servers but some files will probably be available everywhere. Some of those include METAREA_I_WARNING, METAREA_I_HIGH_SEAS_FORECAST and METAREA_I_OFF_SHORE_FORECAST and these originate from the World Meteorological Organization, these and similar for other areas can be seen at http://weather.gmdss.org

Below is an image of what the warning file looked like when writing this chapter.



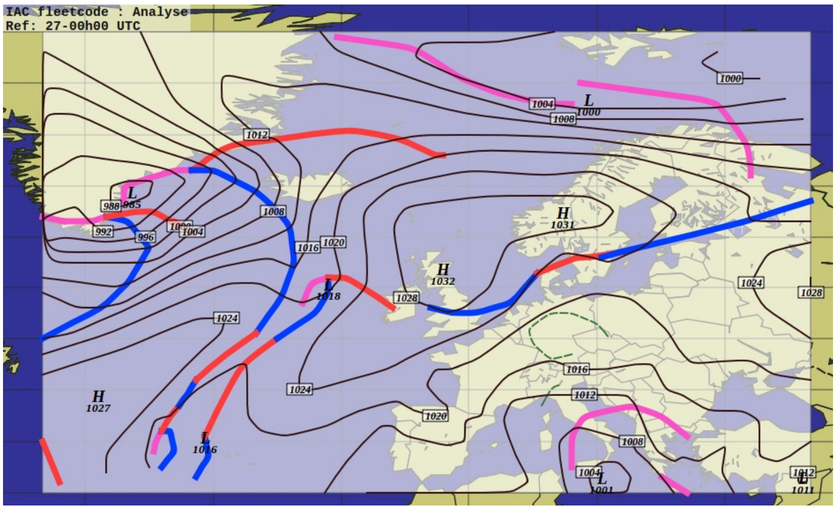
```
METAREA_I_WARNING  - Wed Apr 27 10:07:06 UTC 2011


 WONT54 EGRR 270800
 SECURITE





 STORM WARNING
 AT 270000UTC, LOW 60 NORTH 56 WEST 984, EXPECTED 57 NORTH
 42 WEST 993 BY 280000UTC. WINDS ARE EXPECTED TO REACH
 STORM FORCE 10 IN THE FAR SOUTHWEST OF WEST NORTHERN
 SECTION FROM 280300UTC
```

Getting IAC Fleetcode data

WMO FM 46, or IAC Fleet, are weather standard codes maintained by the World Meteorological Organization. These codes are designed for extreme brevity with mechanical typewriters or telegraphy operators in mind. The standard code is also very good in that its not a standard unique to only one country, language barriers can be overcome by using this common reporting standard.

These codes are indeed very compressed, the image further below came from a file with a size of only 3 kB. The codes themselves are sent as five number code groups and look like “66450 57368 56367 54375 51388 49416 47448 46505...”

As can be seen above these codes are not easily read and to view them it is possible to plot and draw by hand but tools are available that do this. The image below is where this data is presented within the **zyGrib** software (which is free and available for multiple platforms).



To get an updated IAC fleetcode file just connect to a server.
Locate the menu option “Get IAC Fleetcodes” within the **Info** menu and use it. That will have the server download a fleetcode file to your client.

When done just close the connection and open the IAC fleetcode text file using an appropriate viewer (zyGrib for instance). The file is located within ~/.pskmail/Downloads.
For users of other operating systems that “~” means the users home folder. On windows that could be something like: c:\documentsandsettings\users\per\pskmail\Downloads. Main thing is that its stored in your home folder.

Its also possible for the servers to broadcast this file on a regular basis, that way it will effortlessly just be there on the client without the need for a connect. How, when and perhaps even if the servers should broadcast is being discussed and investigated at the moment so this will have to be updated later. Broadcasts could be done at a different frequency, to minimize QRM to the APRS channel.

Getting grib files

Grib file (Gridded Information in Binary) are files used to store forecast information by National weather/meteorology Services. These files contain much more than the IAC Fleetcodes and so the files can be much, much bigger. But, they are also by most considered very valuable as they contain weather information (wind direction and speed) directly usable for navigation. Many electronic chart display systems can import these files and use them together with boat data to optimize the actual boat route.

What you have to do to get them is the following:

* connect to a server

* once connected go to the Info menu and hit 'Get Grib File'

The client will send an email to query@saildocs.com requesting the data, centered on your position +- 5 degrees. For that the server needs to know your return address (send your mail record to the server if that has not been done) The grib file is delivered to your internet mailbox immediately.

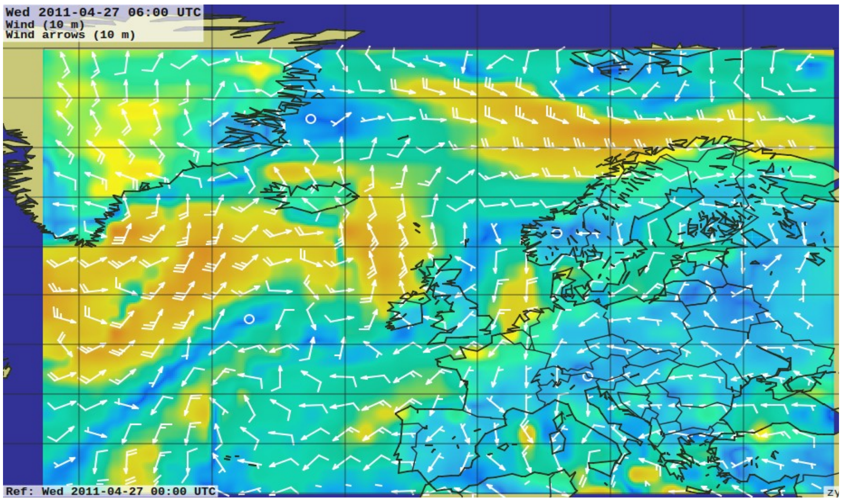
* While still connected, hit the QTC button on the client (return email address has to be correct)

* The mail headers will be downloaded

* Click on the header containing your grib file

* The server now sends the grib file to your client (using ARQ, so no errors)

* When download is ready you can view the attachment with zyGrib.



So, summing it up:

Connect the the server, request the grib file, download the grib file and view the grib file with zyGrib or another viewer for grib files. The attached file will be placed within the Files folder within the pskmail directory (~/.pskmail/Files on linux).

Appendix A, Mode selection

This table tries to gather data about the data modes that are usable with pskmail. Its important to note that this is an attempt and a work in progress, test results and data that can be used to enhance this table is most welcome!

Anyway, a few facts and conclusions can be drawn from this table:

–The fastest mode is PSK500, live tests have shown speeds of 2460 char/min with ARQ.

–The fastest mode for US hams on HF is PSK250

–The slowest mode here is THOR 4 at 148 wpm resp

							Choice (1-3) for				
Mode	**Baud rate**	**Speed wpm capital letters**	**Speed wpm small letters**	**Speed* bps**	**Bandw. (hz)**	**SNR at least**	**NVIS**	**Weak signals**	**High Speed**	**Flutter, fading**	**Description and use**
PSK63	63	74	102	88	63	-2	3	3	3	3	Vy narrow bt also vy slow
PSK125	125	148	204	176	125	-1	3	3	3	3	Better
PSK250	250	296	408	352	250	-1	3	3	2	3	Good all round mode
PSK500	500	592	816	704	500	2	3	3	1	3	Very fast, best 4 good condx
PSKR125	125	74	102	88	125	-8	2	2	3	2	
PSKR250	250	148	204	176	250	-5	2	2	2	3	
PSKR500	500	296	408	352	500	-3	2	3	1	3	Best allround mode?
MFSK16	15,6	58	58	58	316	-8	1	1	3	1	
MFSK32	31,25	120	120	120	630	-6	1	1	2	1	
Dominoex 11	10.8	70	70	70	262	-7	3	2	3	3	fldigi implementation flawed?
Dominoex 16	15.6	100	100	100	355	-6	3	2	3	3	fldigi implementation flawed?
Dominoex 22	21.5	140	140	140	524	-5	3	2	2	3	fldigi implementation flawed?
THOR8	7.8	28	28	28	346	-6	1	1	3	2	
THOR11	10.7	40	40	40	262	-5	1	X	3	2	
THOR16	15.6	58	58	58	355		1	X	3	2	
THOR22	21.5	78	78	78	524		1	X	2	2	

* Speed is an attempt. Varicode and different ways to measure makes this difficult
** SNR is whatever I could find, most at 3 Khz bandwidth..

You may notice that certain popular modes are missing from this table. Two of the missing modes are Olivia and MT63. Olivia is not usable with pskmail as it doesn't support the full 8-bit character set, that means the <SOH> frame start and <EOT> frame end cannot be transferred.

Mode tables for adaptive asymmetric link control

As from jPSKmail version 0.5, separate adaptive mode control for up- and downlink modes is automatic.

To accomplish this a number of changes to server and client were necessary. John, VK2ETA has played an important role herein. The mode table can now be set in the client, and the server will follow this table. Separate RX and TX modes and speeds can be set for the connect sequence, the server will change the modes according to the channel condition.

The table incorporates the most popular modes, covering a large range of speeds, sensitivities and robustness (QRM, Doppler, multipath).

Possible mode table, the modes implemented in jPSKmail client v2:

PSK500 – PSK500R – PSK250 – PSK250R – PSK125R – MFSK32 – THOR22 – MFSK16 – THOR8 – THOR4

The user may select a subset from this list. This subset is actually used in his communication. See page 28.

The modes in these tables run from fast to slow, and become more robust and sensitive from right to left. The server knows the S/N ratio and the download success rate in the client, and chooses the modes which allow the best throughput for the present channel conditions. Initial tx and rx modes are set on the client prior to issuing a connect frame. This is (still) under operator control.

Before sending a frame the server decides if a mode change is possible (up) or necessary (down) based on the S/N ratios of both links and arq success history. In that case the server switches one position up or down in the table.

APRS Mode choice

For the APRS functions the best mode is normally the default channel mode. In Europe this is PSK500R. US servers will use PSK250, unless they operate on MARS frequencies.

In case a station does not come through in the default mode it is possible to use a lower speed mode with RSID on. On receipt of a QSL from a server the client's TXID will automatically be switched off again. This to prevent mode chaos on the frequency...

Squelch Control

The squelch is normally set to 10%.

Without squelch operation the DCD function does not work, and the client cannot send beacons if the squelch is open. DCD means “Data Carrier Detect” and and represents the status if anything is heard.

Appendix B: PSKmail ARQ protocol specification (3rd draft)

Introduction

PSKmail is an ARQ data transmission system. It has been optimized for use on short wave (HF) radio frequencies. PSKmail has both a client/server architecture and a client/client architecture. The servers are gateways to diverse services on the internet. The clients enable mobile or portable radio stations to access the internet where there is no other coverage, or send messages and files to each other.

To use PSKmail, one uses a short wave radio and a PC with a sound card, and often a hardware interface between the radio and the PC. The communications protocol has been optimized for the specific conditions on short wave:

- Varying propagation conditions on short wave, with multi-path, doppler effects, and often weak signals near the noise floor.
- Often the mobile stations run low power and compromise antennas.
- As ham radio uses shared frequencies, interference levels are high.

To cater for this wide range of conditions, PSKmail does not use a single digital radio mode. Instead it uses a range of modes with **separate adaptive control on up- and download channels**. The maximum bandwidth of the HF system is 500 Hz, and speeds range from 44 CPS for ideal channels down to 2 CPS for the most problematic ones. For clean (single hop) channels PSK modes are used, for fuzzy channels MFSK and THOR modes are employed, including convolutional coding, Viterbi decoders, and interleavers to counteract static noise...

Radio interface

The radio is always used in USB mode, as several of the modes used are polarized. THOR modes, e.g., are derived from DominoEx. Often a hardware interface is needed between modem and radio, to provide galvanic isolation for audio signals and PTT signal. *The protocol timing also allows the use of VOX.*

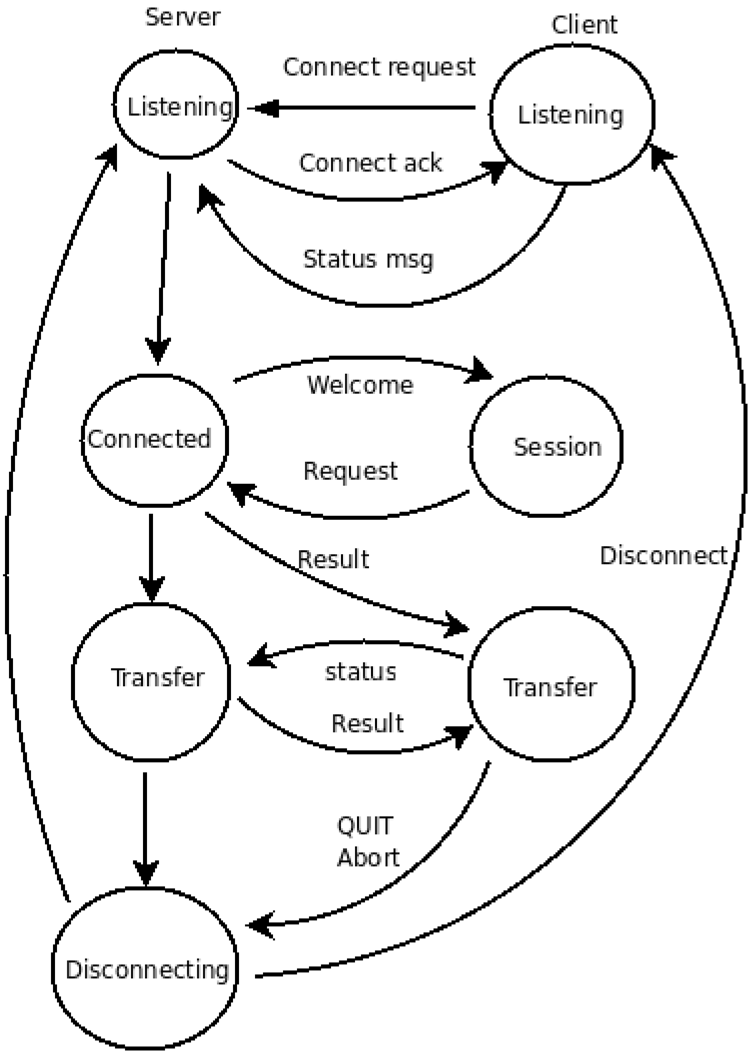
Modem interface

The PSKmail client contains its own modems for PSK, PSKR, MFSK and THOR. PSKmail uses ASCII characters 0x01 (<SOH>, 0x04 <EOT>, 0x06 <ACK>, 0x12 <DC2> and printable ASCII characters 32 ... 127 to talk to the modem.
The system uses **RSID signalling** to notify the link layer protocol engine of mode changes, and also makes signal/noise ratio data available to the application.
PTT signalling is generated by the client itself.
The client program is available for all operating systems.

Link layer protocol

The PSKmail system uses a master/slave protocol. The server is the master, the client is the slave. In case two clients connect each other for a TTY (chat) session, the caller is the slave.
All control functions are in the master, which makes timing a lot easier than in the first PSKmail implementations, where client and server would compete until they were transmitting synchronously on top of each other.

Status diagram



Protocol details

All transactions between server and client are packed in **frames**. Every frame consists of maximum 18 **blocks**, viz. 0 ... 16 data blocks, 1 status block, and 1 optional identification block (at start of session and every 10 minutes thereafter).

Frame structure:

<SOH>[<ident block>]<data block>... <data block><status block><EOT>

All blocks have the same structure. They contain a header, a data partition, and a CRC16 number. The start of a block is marked with a <SOH> (ASCII 0x01) character. A status block ends with a <EOT> (ASCII 0x04) character. The data partition can be **0 ... 64 characters** long.

Block structure:

<SOH><header><data><CRC>
or
<SOH><header><data><CRC><EOT> (status block)

The **header** consists of 3 characters. The meaning of these characters is different for master (server) and slave (client), and also depending on block type. For the **client header:**

Block type	Character 1	Character 2	Character 3
Status	Rx s/n [0x20 - 0x7A]	Session [0x20 - 0x5F]	s
Data	0	Session [0x20 - 0x5F]	Blocknr. [0x20 - 0x5F]
Connect	Protocol version [0,1]	0	c
Summon	Protocol version [0,1]	0	n
Abort	0	Session [0x20 - 0x5F]	a
Ident	0	Session [0x20 - 0x5F]	i

For the **server header:**

Block type	Character 1	Character 2	Character 3
Status	Next TXmode [0 -8]	Session [0x20 - 0x5F]	s
Data	0	Session [0x20 - 0x5F]	Blocknr. [0x20 - 0x5F]
Connect ack	Protocol version [0,1]	Session [0x20 - 0x5F]	k
Disconnect	0	Session [0x20 - 0x5F]	d
Ident	0	Session [0x20 - 0x5F]	i

Some blocks have a **fixed data field**:

Block type	Data partition
Connect	PI4TUE:24 PA0R:1024 xyyyy (x = 0 – 8, initial server TX mode) (yyyy = mode table)
Connect ack	PA0R:1024 PI4TUE:24 x (x = block data length)
Summon	PI4TUE:24 PA0R:1024 10146000 x (x= 0 – 8, initial server TX mode)
Ident	PA0R de PI4TUE

ARQ control and repeats

The receive buffer is a circular buffer with room for 64 blocks. The **status block data field** carries the **status of the receive buffer**:

Char 1	Char 2	Char 3	Chars 4-12
Last block sent this frame	Buffer o.k. until/incl. block x	Last good block received	Missing blocks
[0x20 - 0x5F]	[0x20 - 0x5F]	[0x20 - 0x5F]	0 -8 x [0x20 - 0x5F]

Missing blocks are repeated by adding them to the next frame.

Adaptive mode control

The server controls both its own digital mode and that of the client.

The mode table covers a broad range of speeds and modes:

PSK500	**PSK500R**	**PSK250R**	**PSK125R**	**MFSK32**	**THOR22**	**MFSK16**	**THOR8**	**THOR4**

For use in USA , where Baud rates over 300 Baud are not allowed on ham radio frequencies below 30 MHz, an alternative mode table is available:

PSK250	**PSK250R**	**PSK125R**	**MFSK32**	**THOR22**	**MFSK16**	**THOR8**	**THOR4**

Both client and server measure S/N ratio of the frame received, and the client sends its SNR data to the server in every status block. The server signals mode changes to the client via RSID and through a flag in its status block header.

If a status block is not received, a poll frame is sent, which is just an empty frame without data blocks.

TX and RX modes are controlled in such a way that the number of ARQ repeats are limited, providing optimum throughput for the relative channel.

The server uses RSID on receive, so the client can connect in any of the available modes. Default listening mode is PSK500R, but the server can scan more sensitive modes for DX if necessary.

Mode indicators for sending the Mode table to the server:

1	2	3	4	5	6	7	8	9	a	b	c	e
THOR 8	MFSK 16	THOR 22	MFSK 32	PSK 250R	PSK 500R	PSK 500	PSK 250	PSK 125	PSK 63	PSK 125R	MFSK 64	THOR 4

Squelch Control

Squelch level is automatically adjusted to a proper value by the client, to match the relevant operating mode.

Compression techniques

The PSKmail server uses gzip to compress web pages, emails and binary files. This zipped image is then Base64 coded for transmission with the ASCII127 character set. Coding and decoding are automatic.

Unproto (non-ARQ) operation

There are 4 PSKmail functions which use unproto frames. These functions are:

- Ping
- Inquire
- Link to server
- APRS Beaconing and Messaging

Like ARQ frames, unproto frames consist of a header,a data field and a CRC.
When the client sends a **Ping or Inquire frame**, all servers answer with a **Ping reply frame**. The Ping reply frame contains a signal quality indication representing the RX signal quality in % at the server.

Ping frame	<SOH>00uPA0R:7 ABCD<EOT>
Inquire frame	<SOH>00uPA0R:8 PI4TUE ABDD<EOT>
Ping reply frame	<SOH>PI4TUE:71 99 BCDA<EOT>

Every PSKmail server has a **link table** containing client calls. A client can only be in the link table of 1 server. This function is used when the server sends an APRS message to a specific client. This way, the message is only sent by one server.
The client sends a link request to a specific server, and on receipt of the request the server acknowledges the request:

Link request	<SOH>00uPA0R><PI4TUE ABCD<EOT>
Link acknowledge	<SOH>00uPA0R<>PI4TUE ABCD<EOT>

The server and the client can send a **beacon on HF**, containing its APRS position and a status message:

Server beacon: Client beacon	<SOH>00uIS0GRB-3:72 Pskmail_Server_0.9.34-10147.0KHz -DD6E<EOT> <SOH>00uIK2YXT:26 !4507.95N/01044.65E-I1B80<EOT>

When the server hears a client beacon or an APRS message, it confirms reception with a short **QSL message.** The QSL message contains a signal quality indication (%):

QSL message	<SOH>QSL PA0R de PI4TUE 99 555F<EOT>

An **APRS message** has the same format as the client beacon, an **APRS email** starts with an address@something.xxx .

APRS message APRS email	<SOH>00uDJ0LN:26 PA0R This is the APRS message 1B80<EOT> <SOH>00uDJ0LN:25 PA0R@ISP.ORG This is the email message 2A97<EOT>

TWITTER	<SOH>TWEET This is the tweet 02B2<EOT>

CQ	<SOH>00uPA0R:27 CQ CQ CQ PSKmail 02B2<EOT>

RSID signalling

RSID (Reed Solomon Identification) is used for mode change signalling. Thereby the following rules are applied:

	TX RSID	**RX RSID**
Client Idle (APRS)	OFF	OFF (manual switch ON)
Client receives acknowledge	OFF	OFF
Client connect request	ON	ON
Client data transfer	OFF	ON
Server Idle (APRS)	OFF	ON
Server sends acknowledge	ON	ON
Server data transfer	OFF	OFF
Server changes mode	ON	OFF

The client uses TXID ON more often on slow modes like THOR8 and THOR4.

This to enhance the robustness of the mode change protocol.
Also block size and blocks per frame are lowered using the slow modes.

Appendix C: The hamlib CAT program (rigctl)

Hamlib is required if you want to use the CAT capabilities of your rig. jPSKmail uses the functions

- set frequency
- set PTT on/off (soft PTT)
- select a memory (not yet implemented)
- read frequency (not yet implemented)

hamlib is available from *http://sourceforge.net/apps/mediawiki/hamlib/index.php?title=Download*

There you will also find instructions. Some linux distributions offer hamlib in their software administration system.

When you enter **rigtl -l** in your **console** resp. **command line** and get a long list of rigs, you were succesfull. If you don't know, what a command line is: join a computer class 😉.

References

MFSK modes	http://www.qsl.net/zl1bpu/MFSK/
PSK modes	http://aintel.bi.ehu.es/psk31.html
Fldigi descriptions	http://www.w1hkj.com/FldigiHelp-3.12/Modes/psk.htm
	http://www.w1hkj.com/FldigiHelp-3.12/Modes/mfsk.htm
	http://www.w1hkj.com/FldigiHelp-3.12/Modes/thor.htm
Fldigi	http://www.w1hkj.com/beginners.html
PSKmail:	http://www.pskmail.org/
Mailing list:	http://www.freelists.org/archive/pskmail
hamlib (rigctl)	http://sourceforge.net/apps/mediawiki/hamlib/index.php

REV.8 Authored by: Per Crusefalk, SM0RWO & Rein Couperus, PA0R, 31-01-2013