HRRM v2.0 User Guide

Table of Contents

1 - Background, Concepts and Philosophy....4
1.a - Experimental Techniques....4
1.b - Ham Radio plus Internet....5
1.b.i The conventional approach....5
1.b.ii De-centralized distributed peer to peer networks....5
1.b.iii Listen only stations....5
2 - HRRM via Ham Radio....6
2.a - Downloading HRRM....6
2.b - HRRM Configuration....6
2.b.i Required Information....6
2.c - Fldigi and JS8Call Configuration....7
2.c.i HRRM Fldigi Configuration....7
2.c.ii HRRM JS8Call Configuration....7
2.d - Starting HRRM....8
2.d.i HRRM using Fldigi....8
2.d.ii HRRM using JS8Call....8
2.e - Sending Your First Form with HRRM....9
2.e.i Choosing the Form to Send....9
2.e.ii Selecting the Recipient Stations....9
2.e.iii Filling In the Form Top Section....9
2.e.iv Filling In the Form Main Form Section....9
2.e.v Sending the Form....9
2.f - Ham radio email....10
2.f.i Sending emails....10
3 - HRRM via Ham Radio - Advanced Features....11
3.a - Critical Messages....11
3.b - Parallel message delivery....11
3.c - Relaying Messages....11
3.c.i Basic Process....11
3.c.ii Relay Path....12
3.c.iii Initial Message: List of Message IDS for destination station....12
3.c.iv Relay Message Pull Request....12
3.c.v Relay Node Channels....12
3.c.vi Message Priority....12
3.c.vi.i Critical and High Priority Messages....12
3.c.vi.ii Normal and Low Priority Messages....13
3.d - Peer to Peer Messages....13
3.e - Send Message Options....13

3.f - Dynamic Content Macros....14
3.g - Import & Export - Clipboard Copy & Paste....15
3.h - Customizing Colors....15
3.i - Customized Table Control....15
3.j - Multimode Digital....16
3.k - Sequences....16
3.l - Command Line Options....16
3.l.i Winlink Integration....16
4 - HRRM via Distributed Hybrid Peer to Peer IP Network....17
4.a - Example Scenario....17
4.a.i TCP/IP Model Layers and Resilience....19
4.b - Distributed Hybrid Peer to Peer Network....20
4.b.i Hardware & Service Requirements - How to get started....20
4.c - Configuration options....20
4.c.i Ipv4 and ipv6....20
4.c.ii Ipv4....20
4.c.iii Ipv6....20
4.c.iv Additional notes....20
4.c.v Local Configuration Testing....21
4.c.vi Starlink Satellite....21
4.c.vi.i Starlink Satellite bypass mode with secondary router – most resilient....21
4.c.vi.ii Starlink Satellite + 3rd party VPN – less resilient....21
4.c.vii Terrestrial ISP Internet....21
4.c.vii.i Terrestrial ISP Internet – less resilient....21
4.c.vii.ii Terrestrial ISP internet + 3rd pary VPN – least resilient....22
4.c.viii IDs – Callsign, GUID, LUID....22
4.c.viii.i Ham Radio Callsign....22
4.c.viii.ii LUID....22
4.c.viii.iii GUID....22
4.c.viii.iv Re-Creating IDs....22
4.c.ix Message IDs....22
4.c.x Station Options....22
4.c.x.i Home Network – DMZ / Demilitarized Zone....22
4.c.x.ii Listen Only / Non Ham Radio Stations....23
4.c.x.iii Ham Radio Stations....23
4.d - Bootstrapping....23
4.e - Peer to Peer Distributed Email....24
4.f - Important Considerations....24
4.f.i Subscriptions....25
4.f.ii Additional Instances....25
4.f.iii Data Replication....25
4.f.iv p2p ip station verification....25
5 - HRRM Station Roles....25
5.a - Message Relay Station: Store and Forward....25
5.b - Communications Hub Station....26
5.c - Ad-hoc Gateway Station....26
5.d - Message Repository Station....26

5.e - P2P VHF + P2P HF + p2p ip Satellite....26
5.f - Ad-Hoc Email Gateway....27
5.g - Message Beacon / Digipeater....27
6 - Appendix 1 - Critical Message Communication Techniques....28
6.a - Single Station to Single Station (Peer to Peer)....28
6.b - Single Station to Multiple Stations (Peer to Group)....28
6.b.i Reducing Re-Transmits: Duplicate Fragments....28
6.b.ii Pulling Missing Fragments....29
6.b.iii Send then Pull....29
6.b.iv Pending Messages Notification....29
6.b.v Requesting Message from Stub....29
6.c - Receiving Multiple Simultaneous Messages (Group to Peer)....29
6.d - Multiple Stations to/from Multiple Stations (Group to Group)....30
7 - Appendix 2 - Real-Time Simultaneous Parallel Communication....31
7.a - Multiplier Effect....31
7.a.i Scenario 1a: Gateway....31
7.a.ii Scenario 1b: Real-Time Simultaneous Parallel Communication....31
7.a.iii Scenario 1 Comparison....32
7.a.iv Scenario 2a: Gateway....32
7.a.v Scenario 2b: Real-Time Simultaneous Parallel Communication -Single Channel....32
7.a.vi Scenario 2c: Real-Time Simultaneous Parallel Communication -Multi Channel....32
7.a.vii Scenario 2 Comparison....32
7.a.viii Comparison Summary....32
8 - Appendix 3 - Single Mode Digital vs Multi Mode Digital....34
8.a - Digital Net Setting....34
8.b - Free-for-all....34
8.c - Single Mode Digital....34
8.d - Mixed Mode or Multi-Mode Digital....34

Author: Lawrence Byng

Document Version: 2.0.5 alpha

Revision Date: March 8th 2025

Publication Date: February 21st 2025

1 - Background, Concepts and Philosophy

The traditional top down highly centralized EMCOMM approach works well for emergency communications, however a ground up or grass roots decentralized approach is becoming increasingly relevant. This is sometimes referred to as ‘emergency communications for preparedness’ or emcom-fp for short. Although EMCOMM and emcom-fp share many common facets, the two methods of thinking about emergency communications are fundamentally different. emcom-fp is also not encumbered by EMCOMM dogma. emcom-fp is the driver behind the new feature sets in HRRM.

HRRM started out as a unique and experimental project in the form of SAAM-MAIL and its twin the SAAMFRAM protocol. The short whitepaper in the SAAMFRAM folder on github describes some of the initial inspiration for this project. One of the original main goals was to develop an extremely efficient and resilient protocol for sending critical message forms to stations over the radio in a group setting. Efficiency and resiliency are recurring themes throughout HRRM.

As the project developed over the course of several years through participation in ARES exercises along with much brainstorming and evolution of ideas, other feature sets were incorporated:-

- Use of the form mechanism as a basis to derive other feature sets along with additional form templates for exanple Email and Bulletins
- Development of an experimental data flec mechanism that functions similarly to the UDP internet protocol but over ham radio. This can be extremely effective for notification of critical information over the radio.
- Ability to forward emails and forms to the internet using pat winlink via telnet
- Inclusion of a distributed p2p ip network layer for the integration of ham radio with ip networks including satellite systems such as starlink as well as the terrestrial internet. This approach benefits from high levels of resilience, scaleability, cost effectiveness and performance.

All of these approaches are put together in the HRRM v2 software. The software is in perpetual beta release with open timelines. The software is free and will always be free, it is open source and cross platform with pre-built binaries for raspberry pi, Windows and Debian / linux mint as well as full python source code to facilitate use on many other platforms.

I am always interested to hear ways in which the project can be improved. Please feel free to send me an email or exchange ideas via my github page at gh42lb. Thank you for your interest in HRRM.

1.a - Experimental Techniques

Some of the key experimental techniques used in HRRM are:

- Critical Communications: Fragmentation of a message into checksummed segments that are used to verify acuracy and initiate resends if required.
- Group mode ARQ: Large and complex sets of data can be sent to multiple stations simultaneously and **automatically** error corrected by going round robin through the group and resending only the portions that failed the checksum verification. This maximizes the effectiveness of data transmissions and resends.
- Content only delivery: With no extraneous data repeats or XML or any other inefficient techniques, messages can be sent in a fraction of the time required for other techniques.
- Connectionless communication: Round robin group mode ARQ requires a connectionless process for message exchange and this is radically different from the connection oriented ARQ based modem technologies that allow data transfer with only one station at a time.

Other experimental techniques include the use of :-

- Interleaving or interlacing: Each fragment is sent multipl times at different postions throughout the message to enhance resilience.

- ICS forms over JS8. JS8Call integration is used to send forms such as ICS forms over the radio. This technique is successful however the speed limitations of JS8 present upper limits on what can realistically be achieved when sending data over JS8. Sending ICS forms over JS8 is absolutely viable. Depending on the complexity of the data and mode used, transmission times of around 3 minutes and above compare very well with other transmission techniques used for tough propagation conditions.
- Peer to Peer IP. The latest release of HRRM also includes a distributed hybrid peer to peer network layer for resilient satellite and internet based communications.

1.b - Ham Radio plus Internet

Combining ham radio with internet protocol (IP) technologies can be of enormouse value. The question is how to do this effectively to retain the resilience of ham radio while at the same time limiting dependencies on external terrestrial internet components that are prone to failure.

1.b.i The conventional approach

Conventioanlly, this has involved the use of internet ‘gateways’ where a ham radio station is able to forward email over the internet via a gateway. The is a fantastic concept but there are many weak points in this architecture such as the centralized gateway with its internet connection, the email servers, network services such as DNS, having data stored on a centralized server or gateway, and so on. These multiple weak points result in fragility and a lack of resilience. This strategy is viable when there is no internet outage or an internet outage is relatively small and localized. This is essentially the lagacy approach for achieving ham radio based information exchange over terrestrial IP networks.

1.b.ii De-centralized distributed peer to peer networks

Many of these limitations can be overcome or minimized by the use of a de-centralized and distributed peer to peer ip network where there is no such thing as a central server or a central gateway and the data is distributed across the network. This architecture is highly scaleable, highly reliable and fault tolerant in that if a node drops off the network, there are other copies of the data on other nodes as well as other connection points to keep the whole thing running without a glitch. This results in greater resilience, increased performance and is in many ways the perfect solution for a highly resilient ip layer. Combine this with ham radio and the result is a de-centralized and distributed hybrid peer to peer network that can switch effortlessly between ham radio and ip communications. This hybrid approach also offers additional benefits that cannot currently be realized with an ip only solution.

1.b.iii Listen only stations

With this distributed peer to peer ip technology, the possibility for non ham radio stations to participate in the p2p ip communications is now possible. With this in mind, listen only ie non-ham-radio stations are now an absolutely essential component to the hybrid peer to peer network.

For more information on setting up your station for distributed peer to peer ip networking, please refer to chapter 4) HRRM via Distributed Hybrid Peer to Peer IP Network

2 - HRRM via Ham Radio

HRRM is designed primarily for sending fully verified, error corrected forms over ham radio, peer to peer or peer to group. HRRM has multiple optimizations and supports many techniques that make the communications resilient, even in the most challenging conditions.

HRRM is designed to run as an add-on program to either JS8Call or fldigi. The program is very flexible and has been tested with twenty eight fldigi modes and four JS8Call modes varying from the very narrow bandwidth JS8 slow mode at 25Hz wide to high bandwidth 2600Hz wide PSK modes. HRRM is also open source and cross platform allowing it to be used on many computing platforms. It has been tested with the raspberry pi 4b and the Beelink mini-computing platforms, both of which use very little electrical power and are ideal for off-grid and/or mobile setups as well as with Windows, Raspbian and Linux Mint.

HRRM software has many facets and layers but these all trace back to the original goals for resiliency and efficiency. The latest addition to the HRRM software suite is a peer to peer IP network layer.

2.a - Downloading HRRM

The application repository is at https://github.com/gh42lb/HRRM. This repository contains all the necessary files. The most straight forward way to download these is to click on the green ‘Code’ button then to select the ‘Download ZIP’ file option. This will download the latest HRRM files to your computer. If you prefer you can also download the files individually. The above repository is the only approved repository for the HRRM application.

There are three options for download which depend on the chosen computing platform as follows:

1. Download the windows binary file called HRRM.exe from the github repository (link above). The windows binary can be run on the windows platform and has been tested on Windows 10.
2. Download the raspberry pi binary file called ‘HRRM’ from the github repository (link above). The raspberry pi binary has been tested extensively on the raspberry pi 4b using the standard raspbian OS.
3. Download all of the .py files from the github repository (link above). You will also need to install the following modules: FreeSimpleGUI, sys, threading, json, random, getopt, datetime, socket, time, select, calendar, gps, crc. This can be done using the pip3 command for any missing modules e.g. pip3 install FreeSimpleGUI. Please note HRRM is available for python 3 only.

Downloading of templates is optional as the software has them built in. These files can easily be identified as they end with .tpl

There are currently two template files which may be downloaded as follows:

1. ICS_Form_Templates.tpl
2. standard_templates.tpl

2.b - HRRM Configuration

Once HRRM is up and running, go to the ‘My Info’ tab and fill out the information as it relates to your station.

The call sign and grid square are required fields and must be filled out in order to use the application. The remaining fields are not as critical however, if you wish to use the advanced features of the Dynamic Content Macros, these should also be filled out.

2.b.i Required Information

To establish a connection with another ham radio stations, the following fields need to be populated in HRRM:

1) Station callsign. This is located on the Settings tab
2) Station grid square. This is located on the settings tab.

3) Group name. This field is critical and is located on the main panel and is an arbitrarily chosen group name but it must match the group name of the receiving station. Genrally either @HRRM or @GLOBAL are suggestions. If a receiving station notices a communicatiion with another group name and wishes to interact with the station, the local station group name can be changed to match that of the transmitting station. Generally group name should be either a functional grouping or a geographic grouping or a combination of both.

4) Callsign of peer station in the connect to field. When a station is heard on the radio, the callsign is automatically populated in the peer callsigns table. If the callsign is already in this table, simply click on it to populate the 'connect to' field. Alternatively the callsign can be entered into the 'connect to' field manually.

5) Optional: select a timeframe over which stations will be considered still valid and on air. Generally a setting of about an hour is recommended. If a station has not been heard from during this time frame, it will dissappear from the peer stations table...however setting the timeframe to a longer timeframe such as unlimited will show all stations that have been heard even if they are not on air at the time.

Once steps one thru four are complete, the station can interact with other stations over ham radio.

2.c - Fldigi and JS8Call Configuration

HRRM is run in conjunction with either Fldigi or JS8Call.

2.c.i HRRM Fldigi Configuration

If your station is already setup for digital radio, HRRM should work with fldigi out-of-the-box. HRRM uses the xmlrpc fldigi external API to send and receive messages.

To specify the operating mode you can use the --opmode= command line option (i.e. --opmode=fldigi), however, fldigi mode is the default, so it is currently not necessary to specify this if you are running HRRM with fldigi.

To change to xmlrpc port used to communicate with fldigi, use the --fldigi= command line option as follows:

--fldigi=127.0.0.1:7365

the above option will open xmlrpc port 7365 to the fldigi instance running on the local computer 127.0.0.1

To connect with an fldigi instance on another computer over a local network, this can be done as follows:

--fldigi=192.168.53.1:7362

The above option will open xmlrpc port 7362 (primary xmlrpc port in fldigi) to the fldigi instance running on the computer with the ip 192.168.53.1

Currently HRRM supports one instance of fldigi.

2.c.ii HRRM JS8Call Configuration

To use HRRM in conjunction with JS8Call, there are first a few configuration steps needed in JS8Call as follows:

1. On the 'Mode' menu, the 'Auto Reply' option should be checked. This option is necessary for the proper functioning of text transfers between HRRM and JS8Call.
2. On the 'File/Settings/Reporting' tab find the API section and set as follows:
    - TCP Server Hostname: 127.0.0.1 Enable TCP Server API – Checked
    - TCP Server Port: 2442 Accept TCP Requests – Checked
    - TCP Max Connections: 1 (if using HRRM only) or 2 (if using HRRM and JS8-Net concurrently)

When you are ready to send a communication with JS8Call, you will need to set the appropriate mode in JS8Call as required (Slow, Normal, Fast, Turbo) and make sure the TX button at the top right of the JS8Call screen is enabled.

To change the JSON port used to communicate with JS8Call, use the --js8call= command line option as follows:

--js8call=127.0.0.1:2447

The above option will open JSON port 2447 to the JS8Call instance running on the local computer 127.0.0.1

To connect with a JS8Call instance on another computer over a local network, this can be done as follows:

--js8call=192.168.53.1:2442

The above option will open JSON port 2442 (default JSON port in JS8Call) to the JS8Call instance running on the computer with the ip 192.168.53.1

Currently HRRM supports one instance of JS8Call. Please note, HRRM can be used in conjunction with the open source JS8-Net software. A copy of JS8-Net is bundled with HRRM.

2.d - Starting HRRM

The application currently supports one instance of Fldigi or JS8Call operating one radio. Testing has shown that it is possible to run fldigi and js8call at the same time side by side on the raspberry pi by using VOX based approach to control transmit line but this is not currently a supported mode of operation.

HRRM can be started in two different modes that utilize either fldidg or js8call. Both fldigi mode and js8call mode can be used in conjunction with P2P_IP

2.d.i HRRM using Fldigi

Running the application in fldigi mode is done with the use of the opmode command line parameter:

hrrm --opmode=fldigi

Upon start-up, the main window will display controls that are specific to the fldigi mode of operation.

Fldigi can be used for the full range of text and data based communications in HRRM.

2.d.ii HRRM using JS8Call

Running the application in JS8call mode is done with the use of the opmode command line parameter:

hrrm --opmode=js8call

Upon start-up, the main window will display controls that are specific to the js8call mode of operation.

JS8call can be used with many of the different text and data modes in HRRM with some limitations…

- Sending file data is disabled
- Sending images is disabled

The remaining features include group chats, forms, emails without attachments, ip and port information and a whole array of other small chunks of data or data flecs. Please see the section on data flecs for more information.

2.e - Sending Your First Form with HRRM

2.e.i Choosing the Form to Send

Click on the ‘Compose Msg’ tab. In the ‘Category’ table select ‘GENERAL’ and to the right of that select the ‘BULLETIN’ form.

2.e.ii Selecting the Recipient Stations

To the left of the ‘Category’ table there is a table listing the Call-signs heard. One or more of these can be selected and this will automatically complete the ‘To:’ field with a semicolon delimited list of recipient station call signs.

Alternatively, the recipient station call sign can be entered manually in the ‘To:’ field. If there are multiple recipient stations, make sure the call signs are separated with the ‘;’ semicolon character.

Now click on the ‘Compose’ Message’ button. This will open a new pop-up window showing the blank form ready to be filled in.

2.e.iii Filling In the Form Top Section

The top section of the form shows the following information:

1. Form: This is the name of the form selected in the previous step.
2. MSGID: This is the automatically generated ID that will be used to identify the form. The ID is generated from the sending station call sign and the UTC time to the nearest second that the ‘Compose Msg’ button was pressed.
3. Priority: This field sets the priority of the message.
4. To: This field contains the list of call signs that were set in the previous step. If you wish to make any changes this can be done by editing this field. Call signs must be separated using the ‘;’ semicolon delimiter character.
5. Subject: This field is the subject of the message communication similar to the subject field on an email.

2.e.iv Filling In the Form Main Form Section

The main form section is the actual bulletin form. Go ahead and fill out the fields as appropriate. When finished, click on the ‘Post To Out-box’ button at the top of the screen. If you wish to cancel the form, go ahead and click the ‘Cancel’ button.

2.e.v Sending the Form

Towards the top left of the main application there is a field labelled ‘Connect To:’. If this field is empty, it will be flashing red and green. Fill out the field by specifying the call sign of one of the stations that you will be sending the form to. Each of the recipient stations will also need to fill out this field but they should specify your station call sign as they will be connecting to your station to receive the form.

Now go ahead and click on the ‘Out Box’ tab. In the main table you should see the form that you posted to the out-box listed. If you select the message by clicking on it, you will also see a text representation of the form appear in the preview window that is toward the lower part of the page.

There are several buttons on the ‘Out Box’ page which function as follows:

1. View: Clicking on the view button will open a non-editable full colour representation of the bulletin form message that you selected above. You can use this to review the message prior to sending. When done, click on the close button to take you back to the ‘Out Box’ tab.
2. Copy: Clicking on this button will copy the message to the clipboard. This can be used to export the form message to another application such as an email application or winlink or pat winlink or any other application that accepts cut and paste text. The copied message has two parts as follows:
    - The first section of the copied message is the text representation in human readable form that can be used to review the information.

- The second part of the copied message is at the bottom of the text where you will see ‘HRRM-EXPORT=’ followed by the actual communication itself. This export string can be used either separately or in conjunction with the first part described above to export to another application. The form can be imported into HRRM by clicking on the ‘Clipboard Import’ button towards the upper part of the main screen. The import button will import the message by using only the HRRM-EXPORT= string and place the message in either the inbox if it was addressed to the station or in the relay-box if not.

3. Edit: Clicking the edit button will open an editable version of the form with a new message ID. If you wish to make changes you can do so on this form and when done post it to the out-box. Please note that both the original message and the revised message are both contained in the out-box. If you wish to delete one of these you can select it and click the delete button.
4. Delete: clicking on the delete button will deleted the selected message from the out-box.
5. Delete All: Clicking ‘Delete All’ will delete all of the messages in the out-box.
6. Ready?: This will send out a transmission to the recipient station detailed in the ‘Connect To’ field to ask if they are ready to receive the message. This is an optional step.
7. Send: Clicking the send button will send the message.

You can leave the other settings set to their default values. These have been tested and confirmed to work well over HF and give a good starting point.

Go ahead and click the send button. If everything is setup correctly, you will see the text sent to fldigi or JS8Call and the process of transferring the form starts. You will also see ‘In Session’ start to flash at the top right of the main form. This indicates that a message transfer session is in process.

2.f - Ham radio email

Email is essentially just another type of form. The type of form can be used internally by HRRM to facilitate different processing techniques such as forward emils only to a gateway and so on.

2.f.i Sending emails

To send an email, specify the email address as the recipient and also spcify the callsign of the station to receive the email and forward it to the internet. As an alternative, the sending station can specify the wildcard (*) as the receiving station. With this approach, any station receiving the email that has an active internet connection and is configured to forward to the internet will forward the email.

3 - HRRM via Ham Radio - Advanced Features

3.a - Critical Messages

When sending messages over RF, there are many things that may impact the reliability of the message transfer. Various technologies exist within the ham radio realm to address the issue of message transfer reliability such as acknowledgment handshakes (ACK/NACK) and Forward Error Correction (FEC). Simply sending a message out and hoping for the best may work well in many non-critical application, however, there are instances that require precise delivery of the message along with a high level of certainty that the message was delivered correctly without error.

HRRM essentially encapsulates critical content in a secure wrapper that is delivered in multiple fragments. Each fragment contains a checksum to ensure integrity. This approach enables accurate delivery of the message with an efficient error correction mechanism that only requires erroneous fragments to be re-transmitted. All of which happens completely automatically.

3.b - Parallel message delivery

When sending content to a group of stations, the content can be delivered to each station simultaneously i.e. in parallel. The challenge with this approach is ensuring that the message is delivered accurately and error free. To achieve this, HRRM uses the SAAMFRAM protocol that is designed specifically to allow automatic round-robin checksum based error-detection and resends using what is essential a group-mode ACK verification process. Delivery of critical messages to a group can be achieved very efficiently using this fully automatic group message delivery technique for critical messages. The benefits of using this approach are huge:-

for example when sending a file to multipl stations, typically most of the message will be delivered to all stations with a few fragments needing to be resent. This may vary from sstation to station. For simplicity you can think of the message delivery as being one bulk send of the file plus a few shorter fragment resends to ensure each station receives a full, complete and verified copy of the file. All of this happens completely automatically. More details are contained in the saamfram.pdf documentation on github\gh42lb\saamfram.

3.c - Relaying Messages

Please note: Relaying messages over radio is not fully implemented in current release.

Implementation details subject to change.

3.c.i Basic Process

There are four main ways to relay messages:-

1. Send a stub message beacon and wait for a station further along relay path to pull the full message
2. Send RTS (Ready To Send) to the next station along the relay path
3. Push the full message out passively. This can be done using repeated fragments for greater resiliency
4. Actively push the message to a designated relay station using error-detection and resends to transfer the message to the next station along the relay path.
5. Send out an RTS message over the radio and recipients along the relay path use p2p ip to pull the message across the p2p ip network.

Once the message reaches the destination station, optionally a confirmation receipt can be sent out along the reverse relay path or over the p2p ip network to the originating station.

A combination of manual and automatic forwarding can be used in conjunction with beacons, push and pull techniques to propagate the message along the relay path.

3.c.ii Relay Path

Relay path is determined initially as #n hops and then #n+1 hops if challenges arise. As long as the message to be relayed appears on a node in this path, it is flagged as needing to be forwarded on to the next node in the path.

3.c.iii Initial Message: List of Message IDS for destination station

This process starts by sending out a list of message ids for a given destination station. This list is sent synchronously. The destination station is denoted in the transmission along with the station that is relaying the message along the relay path.

Relay path is determined using a least hop algorithm. This algorithm relies on a constantly updated list of relay stations that can be used to relay the data.

There are three stations memorized at each relay node that can be used for relay.

The initial message also contains the hop count for how many hops are required to reach the destination station. As long as a relay node is either directly on this path (#hops) or on a slightly less efficient path (#hops +1) then the message ids are forwarded on to another relay station further along the relay path for a given destination.

Once the list of message ids reach the destination, the destination station compares the message ids with messages already received. If a message id indicates that a message has not yet been received, the destination station can send back a relay message pull request for that given message id. Messages can be pulled via radio or via p2p ip distributed network.

There are multiple notifications that can be sent out. All message of critical priority, all messages of high priority, all messages of normal priority, all messages of low priority.

3.c.iv Relay Message Pull Request

The relay message pull request works similarly to the initial relay message but in reverse; it originates at the destination station and then propagates back along the path #hops and #hops+1 to the original sending station. If any node along this path had a full copy of the requested message, then the request will not be forwarded on any further. Instead, the relay node will forward the requested message out to the destination along the path #hops, #hops+1.

The first time a relay message is requested, the request is sent back along the exact path of the first relay station to notify of the message. This is the most efficient path. If a response is not received back from the path within a fixed period of time, the message is then requested along any path with #n hops. #n hops being the same number of hops as the original notification and may have multiple actual paths. If no response is received after a fixed amount of time the station will request from path #n+1 hops.

3.c.v Relay Node Channels

In order to avoid collisions, relay nodes along a given relay path each use a different channel for forwarding. For example, a message is sent to the first hop/first relay node using channel x. This first relay node then forwards the message on to hop2/relay node 2 using channel x+1 Relay node 2 then forward this on using channel x+2 etc. This process helps to keep stations that are in proximity from stomping on each oters transmission. Additionally, a relay station will first listen to make sure the channel is clear before forwarding any traffic.

3.c.vi Message Priority

3.c.vi.i Critical and High Priority Messages

For Critical messages and high priority messages, a push technique is used to first push the message along the relay path. The path used will be the shortest path only (#n hops). The message should be sent actively using error-detection and automatic resends to transfer to the next station along the path.

If the message cannot be forwarded for any reason...maybe the next hop relay station went offline, then the station will either:-

1. switch to beaconing mode by sending out notification of the message to be relayed via stub messages and beacon for paths #n hops and #n+1 hops. The beacon message will contain the max path (#n hops) as well as the current number of hops. Any station receiving the beacon stub message that sits within the #n+1 path can then request the full message and then attempt to

push the message further towards the destination. The outer limit of the path propagation is #n+2. This keeps the message path focused to a few stations only.

2. Or send the message out passively and change the max path to n+1 to open up more paths for the message transfer.

Once the message reaches the destination station, a confirmation receipt is sent out along the relay path in reverse. This can either be in the form of a CONF message or in the form of an actual reply.

The receipt confirmation message serves to notify all stations in the path that the message was received..not only to provide confirmation of receipt but to also disengage the relay stations from any further attempts to push the message along the path.

3.c.vi.ii Normal and Low Priority Messages

Normal priority and low priority messages use a different technique to forward the messages along the relay path.

These messages rely on beaconing to notify other stations along the path and then message pull from the next station to move the message along the path.

3.d - Peer to Peer Messages

Messages are typically pushed out from the sending station to the peer station. As long as both stations are active and propagation allows, the messages will be transferred. These messages can be sent either as requiring confirmation / ACK or passively with no receipt confirmation being provided.

If the message is not transferred, several backup methods of message transfer can be used to transfer the messages.

1. If the receiving station received a partial message, the receiving station can then request the sending station to resend missing parts of the message
2. The sending station can send out a stub message to nearby stations that notify the receiving station that there is a message waiting to be sent. The receiving station can then request the sending station send the message.

3.e - Send Message Options

Immediately below the row of buttons on the ‘Out Box’ page you will see a row of options that can be used to fine tune the communications as follows:

1. Pre-Message: This checkbox is used to enable the pre-message option. Currently this is used to send out a stub message for any messages that are waiting to be sent in your out-box. This part of the message is post only and is verified by validating the checksum. It is very useful to communicate to the group if there are any messages pending and waiting to be sent out. If there are multiple pending messages, a single message is chosen at random. Higher priority messages are given a higher probability of selection by the selection algorithm.
2. Repeat Message: This option appears only in the fldigi operation mode. It is used to send one or more additional copies of the message and to interlace these duplicate fragments with the original message. This communication technique is described further in the communication techniques section.
3. Repeat Fragments. This option appears only in the js8call operation mode. It is used to repeat up to the first 3 fragments of the message up to three times. These fragments typically contain the message id and receive list.
4. Include Template: This option will send a copy of the form template used to create the message alongside the content of the form message itself. This can be used by the receiving station to view the message if the receiving station does not have a copy of the form template.
5. Fragment Size: This specifies the number of charters in each message fragment. The default is 30 characters. This can be adjusted depending on band/propagation conditions and also depending on the underlying fldigi or JS8call modulation used. In general, if conditions and modulation mode result in minor errors and a small number of re-transmits, then larger fragment sizes

can be used. Conversely if conditions and modulation mode result in a large number of errors and re-transmits, then a smaller fragment size would be more appropriate.

Also above the tabs in the main area of the form you will see additional options as follows:

1. Fldigi Mode: When running in the fldigi opmode, you will see a list of available fldigi modes listed in this combo-box drop-down. The modes are listed from 1 to 28 in the order of speed with mode 1 being the fastest and mode 28 being the slowness. These modes specify the underlying fldigi modulation mode that will be used for the message transfer. The modes available can be filtered by selecting different option on the 'Span' combo-box drop-down.

2. Channel: This drop-down lists the channels that are available. There are 16 channels that correspond with 16 different offsets in the passband. Each Channel is 125Hz wide. 125Hz is the bandwidth of the narrowest fldigi mode available so if the selected modulation mode is 125Hz wide then then all 16 channels are available for use. If the modulation mode uses a wider bandwidth such as 500Hz then this would effectively mean that there are 4 Channels available: i.e. 1, 5, 9 and 13. The other aspect of the channel is that it ensures both the transmit and receive stations stay on the precise offset which is required for reliable signal decodes in fldigi. Channel is also present when running in the js8call mode and performs a similar role. The Channels and offset values are identical in js8call mode and fldigi mode.

3. Send To: This drop-down gives a list of radios at your QTH that the message can be sent to. Currently there should be only one rig (radio) listed in this drop-down which will be the one configured for use with HRRM.

4. Span: This drop-down gives options that can be used to filter the fldigi modes available by bandwidth. For example 'HF – 500' will filter out any modes that have a bandwidth higher than 500 Hz on the Fldigi Mode drop-down.

3.f - Dynamic Content Macros

Dynamic Content Macros are used to pre-fill parts of the form message during the compose message phase of the process. The Dynamic content macros relate to the individual fields on the 'My Info' tab as well as providing hooks to the current date and time in local time as well as UTC or Zulu time. The current list of available dynamic content macros is as follows:

1. %CALLSIGN%
2. %OPERATORNAME%
3. %OPERATORTITLE%
4. %INCIDENTNAME%
5. %DATE%
6. %TIME%
7. %DATETIME%
8. %LOCALTIME%
9. %ZULUTIME%
10. %UTCTIME%
11. %UTCTIME%
12. %GROUPNAME%
13. %GPSLAT%
14. %GPSLONG%
15. %GRIDSQUARE%
16. %QTHLOCATION%

3.g - Import & Export - Clipboard Copy & Paste

At the top right of the main screen, you will see a button labelled ‘Import’. This button enables importing messages directly from the clipboard into HRRM. The message to be imported must contain a ‘HRRM-EXPORT=’ tag along with the form information immediately following the tag for the import to succeed.

The inbox, out-box and relay-box have an ‘Export’ button. This ‘Export’ button can be used to export the selected message and to copy it to the clipboard. The exported message will have two parts as follows:

1. The first part of the exported message is a text representation of the information contained on the form. This is included by default but is optional. One reason that you may wish to include a text version of the message relates to whether the receiving station has HRRM or not. Another would be that the message is being exported to be sent over the internet or other fast connection and bandwidth is not an issue and the text is included for message preview purposes.
2. The second part of the exported message is the export string and this begins with the text ‘HRRM-EXPORT=’. This export string is critical if the message is to be successfully imported into HRRM at the receiving station.

Important Note: When viewing the text version of an exported message, you must use a mono-spaced font such as Times New Roman or Helvetica and also turn word-wrap off. This will enable the text portion to be viewed correctly.

3.h - Customizing Colors

The ‘Colours’ tab enables many of the default colour settings in HRRM to be changed. The process of changing a default colour involves selecting the new colour in the relevant field then clicking the ‘Update’ button. The different options for changing default colours in HRRM are as follows:

1. Main Heading Background and Text. These two selections allow the default colour of the Main Heading used on the forms to be changed. The setting affects only your station as does not change any of the information actually sent with the form.
2. Sub Heading Background and Text. These two selections allow the default colour of the Sub Heading used on the forms to be changed. The setting affects only your station as does not change any of the information actually sent with the form.
3. Numbered Section Background and Text. These two selections allow the default colour of the Numbered Section used on the forms to be changed. The setting affects only your station as does not change any of the information actually sent with the form.
4. Table Header Background and Text. These two selections allow the default colour of the Table Header used on the forms to be changed. The setting affects only your station as does not change any of the information actually sent with the form.

3.i - Customized Table Control

HRRM utilizes many standard ‘Graphical User Interface’ (GUI) controls however in order to provide the functionality for sending forms it was necessary to create a customized table control so that it could be edited. The functionality of the custom control is still being refined. The main aspects of the custom editable table are as follows:

1. Click on table cell. Clicking on one of the cells in the table allows the field to be edited. When the edit is complete, go ahead and click the return or enter key to complete the update of the field. Make sure that focus is in the editable field when you click the enter key.
2. ‘Tab’ key. First click on the table cell to begin editing. Once you have updated the field, you can click the ‘Tab’ key to update the value and move to the next field. This is very useful if there is a large amount of data to be entered into the table.
3. ‘Add Row’ button. When you click on a cell to start editing, a button will appear that allows you to add additional blank rows onto the end of the table as required.

3.j - Multimode Digital

HRRM provides many capabilities such as group chat, ICS forms, Bulletins, Email, File transfer, Images. Each of these may potentially require a different underlying modulation/demodulation to increase efficiency. HRRM interface provides the ability to switch between these different operating modes seamlessly by the use of a tabbed user interface and multiple modulation selection controls.

3.k - Sequences

The idea behind sequences are to provide the capability to change the modulation depending on how many resends have been requested. Sequences are available for file/image transfer.

3.l - Command Line Options

Th following command line options are available:-

- --opmode=fldigi or --opmode=js8call
- --js8call=127.0.0.1:2447
- --fldigi=127.0.0.1:7365
- --instance=<instance name>
- --listenonly=true
- --show=winlink

3.l.i Winlink Integration

To enable these features, HRRM must be run using the command line option - - show=winlink. On the ‘Winlink’ tab there are two tables, one for the winlink inbox and one for the winlink out-box. In order to setup the integration with winlink you will need to enter the location of the winlink inbox folder in the ‘Winlink Inbox:’ entry field and also the location of the winlink out-box in the ‘Winlink Out-box:’ field. Once this information has been entered, click on the ‘list’ button and this will show all the winlink inbox and out-box messages.

Now select a message from either the winlink inbox or the winlink out-box then click edit. This will open a window that contains the information from winlink. You can now fill out the rest of the fields and click post to out-box and it will be posted to the HRRM out-box ready to send

Please note: The Winlink integration tab is able to read both winlink messages on a windows platform as well as pat winlink messages on a non-windows platform.

4 - HRRM via Distributed Hybrid Peer to Peer IP Network

Please note: distributed peer to peer IP networking feature set is experimental

4.a - Example Scenario

Using a scenario to highlight the feature set is a usefull method of explaining how this all holds together.

Imagine the scenario of a widespread internet outage, this could be from a solar flare, an EMP, a botched software update, a cyberattack, global thermonuclear war, grid failure etc etc. In such a scenario, communications will grind to a halt...cell phones will not function, internet will be unuseable and so on.

Obviously in such a scenario, HF peer to peer radio will still work fine, but what about things like radio-internet gateways? Will they still work...the answer is that this is highly unlikely as the gateways themselves are centralized components that rely on other centralized email servers, not to mention the dozen or so DNS root servers without which the internet would grind to a complete halt...and so on. So it is highly likely that many of these components will fail and along with it, the terrestrial internet communications network will cease to function completely.

Satellite internet is a means to achieving network connectivity in such a scenario, but how will you access your email on such a network if the email servers are down? How will you look up your favorite website if the DNS system is down and the centralized servers for your web site are also down?...the answer is you won't be able to.

So how can you communicate via satellite internet in such a scenario? The answer is ham radio combined with peer to peer ip networking using distributed hash tables and a smart routing protocol such as Kademlia:-

References:-

- https://www.scs.stanford.edu/~dm/home/papers/kpos.pdf
- https://pdos.csail.mit.edu/~petar/papers/maymounkov-kademlia-lncs.pdf

One satellite connected station is able to talk to another satellite connected station using peer to peer ip. In reality there may also be some terrestrial internet servers in the satellite chain of things which are external dependencies and there is no guarantee that these may not be impacted by such an outage, however, minimizing the exposure risk to terrestrial internet dependencies rather than eliminating that exposure risk completely is the goal here.

So...In this scenario where ham radio along with satellite internet is still able to function when all else has failed, how do you participate on this hybrid ham radio / satellite network? How do you find other users? What are their ip addresses and ports? The critical components to making this hybrid network a reality are….ham radio, satellite internet and HRRM v2 software.

In order to start up a distributed peer to peer satellite network in this scenario, each user and the p2p network itself needs to be bootstrapped. This is done by one user powering on their satellite connection, getting their public ip address and then sending this over ham radio via HRRM to other stations. The other stations can then use that internet address in conjunction with their own satellite connected stations to form a communications link. These are the intial nodes on the distributed peer to peer network. These node communication links constitute the underlying process by which the distributed peer to peer network itself is bootstrapped. Other stations also do the same and the result is a network of stations that are able to communicate via ham radio and a distributed satellite peer to peer ip network.

In a widespread terrestrial internet outage, there may however still be some pockets of internet devices that can connect within the confines of a certain smaller geographic region (called network islanding). If there are islands of internet and there is at least one satellite station on that internet island, then **all** stations on that island are able to establish connectivitity to the satellite station and other stations beyond via that one satellite station in conjunction with the p2p ip distributed network ...and it gets even better...non-ham-radio stations that have listened in to the radio communications are also able to use this ip information to join the network as well...and having additional non-ham-radio stations participating makes the network of nodes incredibly resilient...and it gets even better still...there need be only a handfull of qrp ham radio stations to span the entire globe;using JS8call, a station can transmit a qrp beacon every 5 or 15 minutes,

the messages are very short and contain all required info for ham stations and non ham stations to maintain a global communications network. Obviously, if the terrestrial internet connectivity is fragmented, there will need to be more ham radio stations in proportion to the degree of fragmentation. With the p2p distributed network, no data is saved on a central server, instead it is distributed across the nodes on the network using hash tables...this creates an extremely resilient and fault tolerant network that can quickly adapt to new nodes being added and other nodes dropping off the network. As long as at least two users\nodes are active, the network stays running. If all the users stop then the network will also stop...however it can be bootstrapped later by using the same process again.

So the critical components of this are as follows:-

1) Starting up your satellite connection and making a note of the public ip and port # and then starting the p2p network service feature in HRRM for your staion.

2) Using HRRM p2p ip features to notify other users that your station is bootstrapped and provide details of how they can connect to your station.

3) Other stations receiving this data and using the information to bootstrap their own station and then also sending on additional ip adddress connection points over the radio for other user to use to bootstrap their stations.

These three key aspects of the bootstrapping of the nodes and of the p2p network itself translate to the following application layer features in the HRRM software:-

1) The public IP address info that you made a note of is entered into the public IP address field at the top of the main page. HRRM is also integrated with Fortigate routers so that this information can be retrieved automatically.

2) The local station starts up the peer to peer node service by first starting the p2pNode itelf via the desktop icon then connecting to that node from HRRM by clicking the ‘connect’ button at the top left of HRRM and then starting the service by clicking the ‘start’ button to the right of the connect button. If the ‘connected’ text turns green and the ‘started’ text also turns green then the node is active. At this point HRRM will attempt to connect automatically to neighbor stations and also refresh the list of neighbor nodes on the main page.

3) Now go to the P2P IP chat page in HRRM and start a discussion group by entering the name for your discussion group then click add to add it to the list of discussions then click on the ‘announce’ button. This wll send details of your chat group as well as the bootstrapped node ip address out to the group.

4) Stations listening to the radio transmission will see this ip address information popuplate in the p2p IP nodes table to the left of the main screen. These other stations should make sure that their station is connected via satellite, that they have started the node, connected HRRM to the node and started the p2p service and the text is also green.

5) The station that received the information in step 4 can then connect automatically or manually by clicking on the ‘bootstrap’ button. This will connect the station via p2p ip satellite link to the other station(s) and the network has now started in the form of a two node network. As other stations complete this process the network builds.

6) It is now possible to interact with the p2p ip chat feature using peer to peer ip only. The radio should be kept on so that the connection node addresses can be periodically sent out over the radio using the beacon feature to allow more participants to join the p2p ip network.

That in a nutshell is the process. It can be extended to sending email and forms and other messages over p2p ip by using the p2p ip features on the mail outbox and inbox. In a future release, there may also be added the ability to share files and images as this is a relatively straight forward enhancement once the underlying technology has been proven.

Yes there are a number of steps to the process, but these are relatively straight forward. With the right equipment and configuration this whole process becomes trivial. The additional equipment required is negligible...your Starlink mini device and a Fortigate router (or other 3rd party devices and service provider)...once your station is configured, the process to get connected and bootstrapped is as simple as starting the software and clicking a few buttons….that’s it!

The benefit of this approach is significantly increased reslience arising from the near total non-reliance on and avoidance of terrestrial internet related dependencies.

4.a.i TCP/IP Model Layers and Resilience

With regard to the TCP/IP model there are four layers; application layer 4, transport layer 3, internet layer 2 and network access layer 1. The higher level internet protocols such as HTTP, SMTP, FTP, DHCP, DNS exist at the topmost application layer 4.

The scenario of a widespread internet outage has many potential causes. Some of the known risks include:-

- Server failure risks from erroneous operating system patches and erroneous critical application software updates
- IP protocol failure risks from cyber attacks such as DNS attacks and server failure
- Physical network fragmentation risks from EMP, thermonuclear war, severed fiber cables, grid failure

In all of these cases, traditional terrestrial internet communications may be impacted significantly, however peer to peer ip communications may remain viable over much of the existing terrestrial internet infrastructure.

Utilizing peer to peer ip with satellite only or with satellite plus terrestrial internet, creates a multitude of additional ip communication pathways that would otherwise not exist; even if the TCP/IP model layer 4 (corresponding to OSI model layers 5,6 and 7) failed completely, peer to peer ip communication would largely be unimpacted and still viable. Additionally, even if the terrestrial internet network infrastructure becomes fragmented due to impacts at TCP/IP model layers 3, 2, or 1, peer to peer ip communication still remains largely viable when used in conjunction with satellite. Peer to peer ip communications are in this regard significantly more resilient than their conventional internet communication counterparts. There are of course caveats regarding how the satellite networks implement their ip communications at the nexus with the terrestrial internet. That said, peer to peer ip communications can utilize a great many ip communication pathways that simply don’t exist with convenrtional internet communications. For these reasons, peer to peer ip should be considered as an immensely valuable additional communications tool for emergency communications.

4.b - Distributed Hybrid Peer to Peer Network

4.b.i Hardware & Service Requirements - How to get started

This all sounds well and good on paper but how does this translate to things that need to be implemented at your QTH or in your mobile setup to take advantage of this?

The following are the main areas that need to be addressed:-

1. Pre-requisites. Obviously with any emergency communicaiton strategy there needs to be a solid basis to your station which includes batteries such as LiFePo4, solar panels, chargers, inverters, distribution blocks, power efficient computers such as raspberry pi4b or beelink, software such as HRRM, JS8Call, Fldigi, a radio and an antenna system.
2. Internet based – IP requirements. Internet connection preferably satellite for resilience but also can include terrestrial internet service. Optional router capable of inbound ip forwarding and DMZ if required for added security.
3. Listen only stations have similar requirements but can suffice with a good receive only radio to receive relevant transmissions...a device as simple as a $25 SDR dongle is sufficient for this purpose.

Thats essentially it. Having the station tested and configured ahead of time is highly recommended. While there are many satellite internet providers, the focus in this manual is on Starlink Satellite as this has been used in the testing of the new HRRM feature set.

4.c - Configuration options

4.c.i Ipv4 and ipv6

HRRM p2pNode service can utilize ipv4 and ipv6.

4.c.ii Ipv4

To utilize ipv4 for the node service specify an ipv4 address in the ‘p2p ip local address’ field on the settings page on the ‘p2p ip – Satellite and Internet’ tab. Typically this field should be set to 127.0.0.1 or 0.0.0.0 or the statically configured ip address for your local computer that is running the p2pNode service. This field should contain the ip address only and not the port.

The port field should also be filled out in the ‘p2p ip Local Port’ field on the same tab. The default setting is port 3000. If you use a different port, this will also need to be changed on the command line options that are sent to the pnpNode program...on windows these are found on the desktop icon for pnpNode then click properties and change the value accordingly.

4.c.iii Ipv6

To utilize ipv6 for the p2pNode service, specify your ipv6 address in the ‘p2p ip local address’ field on the settings page on the ‘p2p ip – Satellite and Internet’ tab. Typically this is set to :: for the loopback interface or the statically assigned ipv6 address for the computer on which the node is running. Having ipv6 as an available configuration option in HRRM provides additional fexibility for configuraing HRRM for use with 3rd party devices such as Stalink satellite internet, Fortigate routers and so on.

4.c.iv Additional notes

Fortigate routers have the ability to use both ipv4 and ipv6 addresses. ‘Dual-stack’ functionaliy is also available for utilizing ipv4 and ipv6 on a single router interface. Typically ipv4 addresses are converted to ipv6 using a process known as translation. Nat64 allows ipv4 to access ipv6 only networks. Ipv4 mapped addresses are denoted with a ::ffff: prefix.

Additional firewall configuration details regarding port forwarding are outside the scope of this document.

4.c.v Local Configuration Testing

For the purposes of testing your local configuration on your local computer, you can start up multiple test nodes using the following commands from a shell script…

```
#!/bin/bash
xterm -e "python3 ./p2pnode.py -i 127.0.0.1 -p 3000 -s 3001 --delay=5"  &
xterm -e "python3 ./p2pnode.py -i 127.0.0.1 -p 3000 -s 3002 --delay=5"  &
xterm -e "python3 ./p2pnode.py -i 127.0.0.1 -p 3000 -s 3003 --delay=5"  &
```

For this test, the main p2pNode service is listening on port 3000 and the new nodes utilize port 3001 thru 3003 respectively. Running these after the main p2pNode is started, after HRRM is connected to the main p2pNode and after the main p2pNode service is started can be usefull for testing your local configuration. Season to taste for your preferred OS and script file preferences. Alternatiely you can replace the p2pnode.py with the relevant p2pNode binary for your OS.

Beyond this, the ability to connect will need proper configuration of your router\satellite which is outside the scope of this document.

4.c.vi Starlink Satellite

P2p ip can utilize celestial internet i.e. satellite connections as well as regular terrestrial or ISP based internet connections. There are several general approaches with regard to the broadbased configuration as follows:

4.c.vi.i Starlink Satellite bypass mode with secondary router – most resilient

The default strarlink configuration does not allow for the configuration of inbound port forwarding as it is based on Carrier Grade NAT technology or CGNAT for short….however, Starlink does allow the user to configure inbound port forwarding by using a secondary router. To use the approach, the Starlink device must be placed into bypass mode and the necessary inbound port configured on the secondary router. This architecture has the least exposure to external dependancies on the terrestrial internet and is viewed as potentially the most resilient configuration for doing peer to peer ip comms in an emergency environment. Having a public ip that changes infrequently is definitely a plus however it is not a requirement... even in the case where the public ip changes each time the satellite connection is started , utilizing peer to peer comms is still viable. Once a satellite connection has been started the public ip must be noted and entered into the HRRM software prior to starting the p2pNode service. If your station also has a Fortigate secondary router, this public ip information can be retrieved automatically using the Fortigate API, which is also built into HRRM, with the click of a button.

4.c.vi.ii Starlink Satellite + 3rd party VPN – less resilient

Out of the box, Starlink can still be utilized for peer to peer communications by using a 3rd party cloud based VPN service such as ProtonVPN or PIA. This allows the user to establish a VPN connection to the cloud and also has capabilities for inbound IP forwarding. Generally, the VPN providers allocate a random port for this so it is necessary to first establish the VPN connection to the cloud and to then locate the public IP and port and enter those into HRRM manually. This information will change each time the link is dropped and restarted and will need to be re-entered into HRRM each time. This method has additional dependencies in the form of Domain Name Services and cloud based VPN service providers so it is not as resilient an architecture as the section above however, it nonetheless still provides an effective method for doing p2p ip via Starlink.

4.c.vii Terrestrial ISP Internet

4.c.vii.i Terrestrial ISP Internet – less resilient

Most home internet routers have the ability to configure inbound port forwarding. This approach requires the user to configure their router and to then make a note of ip address information and to enter both the public IP and assigned port of the router into HRRM. Generally the public IP and port will be relatively persistant but should be checked each time a new internet session is started and HRRM settings update if required. If the router is a Fortigate router, this information can be retrieved automatically and the necessary fields in HRRM

populated. Again, due to increased external dependencies with this architecture, it is considered less resilient for emergency communications.

4.c.vii.ii Terrestrial ISP internet + 3rd pary VPN – least resilient

A terrestrial ISP internet connection in conjunction with a 3rd party VPN such as ProtonVPN or PIA allow the user to establish a VPN connection to the cloud. Generally, the VPN providers allocate a random port for inbound port forwardiong so it is necessary to first establisdh the VPN connection to the cloud and to the locate the public IP and port and enter those into HRRM. This informationm will change each time the link is dropped or closed and will need to be re-entered into HRRM each time. As this approach has the greatest number of external dependencies, it is considered the least resilient.

4.c.viii IDs – Callsign, GUID, LUID

To participate in a peer to peer ip network, each station must have an ID. For ham radio operators, there are three choices of ID; Callsign, GUID or LUID. For listen only stations there are two choices of ID; GUID and LUID.

4.c.viii.i Ham Radio Callsign

For ham radio stations, the option to use the callsign as the ID for peer to peer ip communications is available. This is the recommended choice, as a ham radio callsign is both globally unique as well as being relatively short. Having a shorter length ID is more efficient for message transmission times and reduces data storage requirements.

4.c.viii.ii LUID

The LUID is a locally unique ID or in other words an ID that should be sufficient most of the time but is not guaranteed to be unique. The ID is based on the station's 'nickname' field with an additional 5 random characters joined on the end along with an '*' character. LUID is the default for listen only stations that do not have a callsign.

4.c.viii.iii GUID

The GUID is a globally unique ID based on the mac address combined with a timestamp. This option is also available to listen only stations that do not have a callsign.

4.c.viii.iv Re-Creating IDs

Ids can be recreated on the settings tab. To re-create the GUID or LUID it is first necessary to click the 'unlock' button alongside the respective ID and to then click the 're-create' button to create a new ID. A new ID is then created and populated in the ID field. To make this change permanent, it is necessary to click 'save' or 'save and exit' in order to save the changes.

Please note: Once a station ID is changed, access to content associated with any previous IDs for this station is terminated. Use with care!

4.c.ix Message IDs

Message IDs are a combination of the station ID + a timestamp. For ham radio station IDs the two sections of the message ID are separated with an '_' character. For GUID and LUID based message IDs, the station ID and timestamp are separated with a '#' character.

4.c.x Station Options

4.c.x.i Home Network – DMZ / Demilitarized Zone

The recommended configuration for a home station node is to have your satellite / internet connection connected to a router that supports a Demilitarized Zone or DMZ. This provides additional security for the locally attached network and computers. A firewall that has the ability to setup a DMZ is a requirement for this approach. Configuration of a DMZ on the router is outside the scope of this manual.

A small computer device with the p2pNode software is then placed on the DMZ. This device should contain minimal software. A full backup of the hard drive is also recommended. A raspberry pi is one possible solution for this. Cloning the sd card to another sd card for

backup. In case there is a breach on your network and the device gets compromised, the sd card can be swapped out easily by copying the backup sd card to the card to be overwritten. Then putting one of these back in the device to complete the restore process.

HRRM software should be kept on the internal network and not on the DMZ. Connections are made to the p2pNode using the built in SSL VPN connection from internal secure network to DMZ. Peer to peer connections to the distributed P2P network are made from the DMZ to the external network. An initial hrrm.key and hrrm.cert self signed certificate are provided with HRRM however users can generate their own self signed certificates to use with the SSL VPN connection for additional security if required.

4.c.x.ii Listen Only / Non Ham Radio Stations

Non-ham-radio listen only stations can participate in the p2p ip network by using either a globally unique identifier (GUID) based on their computer mac address and timestamp or a locally unique identifier (LUID) based on the station nickname field + timestamp. Listen only stations are not able to transmit messages over the radio.

Stations that join the p2p ip network that are not licensed ham radio operators are designated as listen only stations in HRRM. These stations can utilize all of the benefits of p2p ip networks while they have a functioning ip satellite or Internet, however if this is not available then the stations can still receive both notification messages addressed to the group as well as read only messages sent directly to them over the radio. To receive these messages, the receiving station needs a copy of HRRM installed along the the p2pNode, modem software to connect to HRRM such as fldigi or js8call and a radio, antenna and battery. The remaining step is to tune into a ham radio digital net such as a js8 net or an fldigi net and to let the software listen. Messages will appear in the chat box and inbox.

Non ham radio stations i.e. listen only stations, use the p2p ip Identifier GUID or LUID. As there can be no guarantees made about an LUID being globally unique, the identifier is combined with the group name of the station for that individual to form the full identifier.

4.c.x.iii Ham Radio Stations

When sending radio messages, the stations group name is used to index all communicatios. For example, the WH6ABC is combined with the stations group name ‘@MYGROUP’ to produce a complete identifier of WH6ABC@MYGROUP’ . Think of this as a subscription to the group @MYGROUP. This identifier is simply a reference to the address space allocation within this groups address space. The station will have a group name assigned which will be used as the stations secondary mail address. When sending messages over the radio to a specific group, you can think of this as all stations as being subscribers to the group...for example a net that uses a group name of ‘@HINET’ would be a communications hub for subscribers to the ‘@HINET’ group. When ham radio stations are sending or forwarding messages via P2P IP, **two** address spaces are utilized...the primary address of <CALLSIGN>@GLOBAL as well as the secondary address of <CALLSIGN>@<GROUPNAME>. Both identifiers are utilized when sending and receiving messages by ham radio stations.

Ham radio stations use the station callsign as a globally unique ID. Think of this as an assignment in a global address space i.e.< MYCALLSIGN>@GLOBAL. When sending messages on the radio or via p2p ip, the @GLOBAL part is implied and does not need to be explicitly stated in the transmissions.

4.d - Bootstrapping

How to bootstrap your station and the p2pip network:

- configure your router so that UDP port 5222 is forwarded to the local network...known as port forwarding. You may need to also open inbound UDP port 5222 in the windows or other firewall. Other port number can be used instead. 5222 is used as an example.
- Start up your satellite or terrestrial internet connection
- Start HRRM application and populate the public IP field in HRRM with your public ip information. This can be done automatically by using the built in fortigate integration to retrieve this from a Fortigate router.
- Start the p2p node application
- Click connect at top left of HRRM to connect HRRM to the p2pip node
- Click start service to start the p2pip srvice

- Enable beaconing. This will send out a beacon containing relevant ip information that will enable other station to bootstrap their stations and connect to you station forming a distributed ip network. HRRM attempts to connect to other stations automatically. If a connection is made, the neighbors button wil turn greem and the list of neighbor ips appear in the display table….you are now online on the distributed peer to peer network!
- Now that the stations are connected and the network is started you can use the p2pip discussion group feature or the p2pip send feature on the outbox to send text and emails. Receiving stations should periodically check for messages using the p2pip get messages buttons.
- As more stations hear the beacon signal, they can join the network thus providing more ip nodes that other stations can use to connect.

The additional equipment required is negligable...your stalink mini device and a Fortigate router...once your station is configured, the process to get connected and bootstrapped is as simple as starting the software and clicking a few buttons….that's it!

4.e - Peer to Peer Distributed Email

Using the above feature set, it is now possible to build a truly distributed email system outside of the converntial web based email. There are many benefits to this approach but these are outside the scope of this manual.

The essential point of this distributed email are:-

- HRRM distributed email is available to ham radio operators by using the callsign as a globally unique identifier. Non-ham-radio listen only stations can participate in the p2p ip network by using a globally unique identifier based on their computer mac address and timestamp.
- Email addresses are of the form <callsign>@groupname for ham radio stations and <GUID>@groupname for the non-ham stations.
- Emails can be shared between other members of the group over the radio and via P2P IP
- Once the emails have reached a p2p ip connected HRRM user, the emails are then able to be forwarded and replicated with other p2p ip HRRM users at much higher speeds than are possible using only radio.

4.f - Important Considerations

Some important aspects of the software are detailed below:-

- When posting a message to the messages outbox, there are two delivery methods available, one for delivery over radio and one for delivery over ip Satellite / Internet.
- Message Ids for radio messages contain an '_' underscore and are based on the stations callsign
- Message Ids for ip Satellite / Internet delivery contain a '#' hash character and are based on the p2p ip GUID
- Radio messages can be sent over p2p ip Satellite / Internet however, IP Satellite / Internet messages ***cannot*** be sent over radio unless the p2p ip message is first edited by a ham radio station and then re-posted for radio delivery by his station.
- Editing of messages creates a new message id.
- Distributed mailboxes have an expiration of 3 months. If no new messages are received by the distributed mailbox in a 3 month timeframe, the mailbox is removed.
- Delivery of messages is not guaranteed. It is important to retrieve messages within the given expiration timeframes.
- In the current release, all messages are sent 'in the clear' without any form of encryption however, the station to station VPN can be used to protect messages going between stations when connected via VPN.

4.f.i Subscriptions

Stations are subscribed to the group specified in their application configuration. When sending over the radio, the group name acts as an exclusive filter that requires any station wishing to communicate with the group be be subscribed to that group.

When sending and receiving messages via p2p ip, stations have the flexibility to communicate across groups for example...fred.smith@group1 can exchange messages with john.doe@group2 .

4.f.ii Additional Instances

Additional instances can be created on the same computer by using the command line option /I<InstanceName>. This will create a new set of configuration files and folders under the new root HRRM\Instance\InstanceName. This is usefull to allow mixed mode communications such as one instance interacting with JS8Call and another interacting with fldigi. Data replication can be used to replicate sent and received data across these instances.

4.f.iii Data Replication

HRRM allows incoming data to be replicated to a secondary instance of HRRM on another network connected computer over an SSLVPN conneciton. This can be usefull to allow a multi computer setup where each computer is dedicated to a particular radio.

Only complete messages are replicated. Any fragments remain only at the station that received the fragments. Once the remaining fragments have been received by that station, the full verified message is replicated.

4.f.iv p2p ip station verification

HRRM can periodically check the state of known p2p ip connections to verify if they are active and still valid. Upon success, the station in the p2p ip list is set to green. The check involves using the IP address and port number currently listed for each station and attempting to ping the node. If successfull, the station is marked as green in the list.

Under settings, adjustments can be made to how frequently HRRM will re-verify the list of p2p ip connections and credentials. This can be set to check on startup and 5, 15, 30, 60 minute intervals.

5 - HRRM Station Roles

In any given situation stations are often assigned a different role for facilitating message transfers. HRRM is very flexible and integrates with multiple other platforms to allow use of a wide variety of communication techniques. HRRM offers additional capabilities that enhance the line-up of techniques available.

Some of the different station roles that can be used in conjunction with HRRM are as follows:

1. Message relay station for store and forward
2. Communications hub station
3. Ad-hoc mesh node station
4. Message Repository station

These different roles are described in the following sections.

Please note: During a disaster situation the internet may not be reliable. If portions of the internet are still usable then the internet based modes described in the following sections might still be viable. However, when utilizing internet based communication during a disaster receiving a confirmation reply via the same communication channels will be necessary to gauge the effectiveness of that particular mode of operation. This would apply to internet connected gateways, regular email and other internet based techniques.

5.a - Message Relay Station: Store and Forward

When a station is operating as a message relay the following HRRM features can be used

1. Relay Box. The relay box will store any messages received by the station that do not include the relay station in the receive list. To forward a message that is stored in the Relay box, select the message in the relay messages table then click on the ‘Add to Out-box’ button. This will queue the message up in the out-box. Now specify the station you wish to send the message to in the ‘Connect To:’ field then click on the out-box tab, select the message that you just queued and then click send.
2. The relay station can listen for requests directly to pull a specific message with a given message ID or requests for which stations have copies of a given message stored. A relay station can respond to both of these requests and either initiate a send or respond to a pull request.

5.b - Communications Hub Station

A hub station typically receives messages from one or more sources and then sends these out.

The HRRM features used by a hub station include:

1. HRRM has the ability to export HRRM messages so that they can be sent via other applications. Any messages can be exported into an export format by clicking on the copy button. This will produce a text version of the message along with a HRRM-EXPORT= tag at the end that contains all the necessary information to reproduce a full copy of the message in HRRM. Any messages received that contain a HRRM-EXPORT= tag can be copied to the clipboard and then imported directly into HRRM by clicking on the ‘Clipboard Import’ button. The message will then be added either to the inbox or the relay box and from there it can be posted to the out-box for sending out via HRRM.
2. Winlink. To utilize this feature follow the section that describes winlink integration in the advanced features section. By using this approach, messages can be sourced from winlink or pat-winlink and sent out via HRRM. Additionally any winlink messages received that contain a HRRM-EXPORT= tag can be imported directly using the clipboard as described above then sent out.
3. Email. Similarly any emails received by the hub station that have the HRRM-EXPORT= tag can be processed in a similar manner to the above.
4. Voice. Stations receiving message via voice can use the ‘Compose Msg’ tab to create a message then post to the out-box to be sent out via HRRM.

5.c - Ad-hoc Gateway Station

Using similar techniques to those already described, it would be possible without any special configuration to set up a station as an Ad-hoc Gateway simply by starting up the HRRM application. The integration with internet based applications currently requires a manual cut and paste approach as described in the earlier sections however this functionality opens up the possibility of setting up an Ad-hoc Gateway to handle message traffic.

5.d - Message Repository Station

A message repository station could be used to store many messages that can be queried and pulled by any other station. There will typically be two types of message in this category as follows:

1. Messages addressed to a group: By using this approach group messages can be stored and retrieved by any member of that group. This could be a very useful method to disseminate information to a group of stations.
2. Individually addressed messages: This is similar to the relay role described above. These types of messages are available for any station to pull as required.

5.e - P2P VHF + P2P HF + p2p ip Satellite

This combination is considered to be one of the most resilient and flexible approaches to establish and sharing digital information in emergency situations. This will require some users to have a hub capability utilizing 2 or more of these communication methods. There

are different techniques to this approach such as switching the radio to different communication modes temporarily and then back. This can also be achieved by using the data replication feature in HRRM to synchronize received data to a secondary station.

5.f - Ad-Hoc Email Gateway

HRRM can be configured to forward emails received via P2P radio to the internet automatically. There are several requirements to achieve this:-

1. Pat winlink is required to forward the emails to the winlink system
2. an active internet connection is also required
3. On the settings tab, check automatic forwarding for emails
4. Specify the location of the pat winlink binary used to forward the email to the internet.

With the above configuration, emails received by the station will be forwarded to the internet automatically.

5.g - Message Beacon / Digipeater

A message beacon will beacon out messages for a group. Only one group can be specified for the beacon.

Messages are sent out to the group and when picked up by the station are then periodically digipeated as stub messages. Stations that receive the stub message can then request the full message from the message beacon. When the message is digipeated from the beacon, both the sender address and the from address are provided in the header . This is to facilitate those within range from easily identifying the source of the message and requesting any missing data from the sender station.

6 - Appendix 1 - Critical Message Communication Techniques

These techniques are used for the transfer of fully verified error corrected messages as would be the case for critical messages.

6.a - Single Station to Single Station (Peer to Peer)

Sending a verified, error corrected message from one station to another using the peer to peer approach is done simply by specifying only a single receive station in the ‘send to’ list when the message is composed.

Peer to peer communications of forms is optimized for shorter message length, faster transfer speeds and increased resilience to adverse band conditions.

6.b - Single Station to Multiple Stations (Peer to Group)

This can be achieved by entering a list of recipient stations in the ‘To:’ field when composing a message. All specified receiving stations will need to be on frequency and listening for the incoming message.

During the send process, the station sending the form will verify with each receiving station in turn that they have received the message in full without error and if not will re-transmit any missing parts. At the end of this process each receiving station should have a fully verified, error corrected copy of the original message.

Please note: during the re-transmit phase, the sending station will make multiple attempts to resend the missing parts up-to a preset number of retries before moving to the next station on the list. If the message cannot be transferred within the specified number of retries, there are additional options available to complete the message transfer process as follows:

1. Following completion of the transfer session, any station that has not received a fully verified, error corrected copy of the message can send out a pull request to the sending station to pull the message.
2. The sending station can make additional attempts at sending the message to the stations that did not verify during the original session using the message push approach.
3. The receiving station can request missing fragments from a relay station.

Please note, if there are a large number or receiving stations specified (i.e. more than 10 or so) on the receive list, then this technique can still be used however another technique may be more appropriate such as that discussed in the next section.

6.b.i Reducing Re-Transmits: Duplicate Fragments

This can be done by sending additional copies of the full message (full set of fragments) or sending additional copies of specified fragments such is the first 3 which contain message ID and receive list.

1. Sending Multiple Copies of the full message. This option is currently available only with the fldigi opmode. The additional message copies are interlaced with each other to increase resilience to any impacts to the send stream. Sending multiple copies of the message may be a good strategy if there are a large number of receiving stations as the extra time to send an extra copy could well be less than the time required to go station by station and re-transmit when only a single initial copy is sent out.
2. Sending additional copies of first 3 fragments. This option is currently available only with the JS8Call/JS8 operating mode. The purpose for doing this is to ensure the first three fragments that contain the most critical information are sent several times. Once received, these initial fragments will provide key information such as message ID and recipient list that the receiving station can then use to determine if a full copy is required and if so to have the message ID that can be used to request the full copy.

6.b.ii Pulling Missing Fragments

This technique can be used if the message transfer was not successful during the original transfer session. There are two approaches to pulling any missing fragments as follows:

1. Request missing parts from sending station. The station that sent out the original message is known to have a full copy of that message so the process is quite simple. If you have a message in the inbox or relay box that shows ‘partial’ then you can send a request to the sending station by selecting the message in the table in the inbox or relay box and then clicking the ‘Request Msg’ button. This will pull the message from the original sending station and attempt to provide a fully verified, error corrected copy of the message using a similar process as before.

2. Request missing parts from a relay station station. In order to use this approach you will first need to send out a query to determine if there are any other stations on frequency that have a copy of the full message or specified missing fragments. This can be done by selecting the message in the inbox or relay box then click on the ‘Query Msg’ button. This will send a query to the group of stations with the message id. If a relay station has a copy and replies back to confirm, then you can initiate the pull request in the same manner as step 1 above, the only difference is that you will change the ‘Connect To:’ field to be the call sign of the relay station. Then click ‘Request Msg’ button and the transfer of missing fragments will begin.

6.b.iii Send then Pull

This technique could be used for a large number of stations. The technique involves sending the message to the group then having members of the group request any missing parts using the pull technique. For this process to be effective it would make sense to send multiple copies of the message in the initial send to reduce the number of missing fragments at the receiving stations.

6.b.iv Pending Messages Notification

Notifying one or more stations of pending message can be done by using the pre-message option. This will send notification for any messages waiting in the out-box or the relay-box. The notification is achieved by sending (posting) a very short check-summed message that contains the message ID and the receive list only. This is sufficient information for any station that receives the stub to determine if they wish to pull the rest of the message.

6.b.v Requesting Message from Stub

1. Inbox. Any station that receives a stub notification message will have a message appear in the inbox or in the relay box with the word ‘stub’ in the ‘completion’ column. Any stub message can be selected and then the ‘Request Msg’ button pressed to initiate the transfer for the full message.

2. Relay Box. If the stub message appears in the relay box, a similar technique can be used as described above. Hen complete the full message will show ‘Verified’ in the ‘completion’ column. Verified messages also appear with a green colouring in the messages list.

6.c - Receiving Multiple Simultaneous Messages (Group to Peer)

The key aspect to Group to Peer communications has to do with the technique of simultaneous parallel communication.

This enables a modem application to be able to listen to multiple different offsets relative to the given frequency at the same time. This technique is used extensively in the WSJT-X and JS8Call applications for FT8 mode and JS8 mode respectively.

If the modem has this ability to listen to multiple channels simultaneously then HRRM is able to utilize this for receiving messages using the passive receive approach from multiple stations simultaneously.

This could be very useful if there is a large amount of message traffic going back and forth between stations. In the event that some parts of the message are not transferred successfully during this passive mode message transfer, the receiving station can initiate a pull request to the sending station or to a relay station as described in the earlier sections.

Please note: Due to current limitations, Group to Peer communications are only available using JS8Call.

6.d - Multiple Stations to/from Multiple Stations (Group to Group)

The key aspect to Group to Group communications also has to do with the modem application being able to listen to multiple different offsets relative to the given frequency at any time. Additionally there will be multiple sending stations spanning one or more groups as well as multiple listening stations also spanning one or more groups. In this way, group to group communication is now possible.

Please note: Due to current limitations, Group to Group communications are only available using JS8Call.

7 - Appendix 2 - Real-Time Simultaneous Parallel Communication

I have coined the term ‘Real-Time Simultaneous Parallel Communication’ in the context of digital ham radio, in order to help explain this technique.

The technique of Real-Time Simultaneous Parallel Communication has the following key components:

1. The ability for a single digital station to receive and decode multiple digital transmissions simultaneously on a single frequency
2. The ability for a single digital station to send digital messages to a specific group of digital stations simultaneously.

This technique has been around at least since WSJT-X / FT8 and JS8Call / JS8 arrived on the ham radio scene. WSJT-X for example can typically decode 30 or more 50Hz wide FT8 stations simultaneously on a single HF (3kHz wide) frequency. The technique was further enhanced by JS8Call with the ability to send messages to a specific group of digital stations. The technique has proved to be extremely popular in the context of making a QSO for station to station communication but has not been applied to any great degree to communications to/from groups of stations i.e. peer to group, group to peer and group to group. Digital nets do exist and are slowly becoming more popular but at present these are more of a niche area that is still evolving.

When utilized for transferring messages among groups of digital ham radio stations, Real-Time Simultaneous Parallel Communication has enormous potential. Included in this section are several examples and scenarios giving details of how performance improvements can be realized. Key to understanding this is something called a multiplier effect as described in the following section.

7.a - Multiplier Effect

To illustrate the multiplier effect, two scenarios are provided for comparison. The first relates to peer to group communications and the second relates to group to peer communications. Each scenario compares the gateway approach to the real-time simultaneous parallel communication approach. The first scenario involves sending a form from one station to five recipients (peer to group). The second involves sending five forms, one from each of five sending stations, to one recipient station (group to peer). In both scenarios, message transfer is achieved using radio based communications only.

7.a.i Scenario 1a: Gateway

To send a form to five recipients, the sending station must first contact the gateway and send the message. Following the initial message upload, each of the recipient stations must then contact the gateway to retrieve the message. The stations have no knowledge that there is a message waiting for them on the gateway so they must check back periodically by connecting to the gateway to see if there are any messages waiting. When a recipient station sees there is a message on the gateway for them, they must then retrieve that message. The same process is also required for the other four recipient stations. At the end of this process each recipient station will have a copy of the message.

To summarize this, the sending station and each of the five recipient stations has contacted the gateway at least one time to retrieve the full message. This effectively ties up this particular channel (frequency + offset) for six separate communication sessions. So for any given communications mode, the total time = upload transfer time + (retrieval transfer time x 5) = message transfer time x 6.

7.a.ii Scenario 1b: Real-Time Simultaneous Parallel Communication

When distributing communications to many stations simultaneously in real-time there are many advantages. Not only does the message get received in real-time so the station does not have to keep checking back, but the message can be transferred more quickly. Using this approach, the sending station will send out the message once and each of the receiving stations will receive that message as it is being sent out. If each of the receiving stations were to receive the message on the first attempt, this will have significantly shortened the amount of radio time required to transfer the message on that channel (frequency + offset). In this case for a given communications mode, total time = message transfer time x 1.

7.a.iii Scenario 1 Comparison

So to compare scenario 1a and 1b above, you can see that scenario 1b is six times faster than scenario 1a. Or to put this another way, scenario 1b can use a modulation mode that is six times slower than that used by scenario 1a and still transfer the messages in the same amount of time with the added benefit that the stations do not have to keep calling the sender to check if there is a message waiting. This is a multiplier effect as it relates to real-time simultaneous parallel communication in a peer to group setting.

This multiplier effect for a group of stations on frequency and listening all at the same time makes this a highly efficient mode for transferring messages to groups of stations.

In addition there is a second multiplier effect as it relates to real-time simultaneous parallel communication in a group to peer setting. This scenario involves five stations sending a single form to a single recipient station:

7.a.iv Scenario 2a: Gateway

Each of the five sending stations must first contact the gateway to upload the message. The recipient station must contact the gateway periodically and then retrieve the messages. Regardless of the fact of weather this retrieval is spread over multiple sessions or all in one session, the receiving station will still be utilizing the air time for five message transfers. So to summarize this the total time that a given channel (frequency + offset) is in use using this approach is total time = (5 x message upload time) + (5 x message download time) = 10 x message transfer time

7.a.v Scenario 2b: Real-Time Simultaneous Parallel Communication -Single Channel

Each of the five sending stations will use a single channel (frequency + offset) to send a single message to a receiving station. Because this is being done using a single channel, each of the sending stations must take turns in using this channel in the same way as if they were sending to a gateway. The receiving station will receive these messages as they are being sent. To summarize this, the total time on air for this transfer of the five messages using this approach is total time = 5 x message transfer time.

7.a.vi Scenario 2c: Real-Time Simultaneous Parallel Communication -Multi Channel

Each of the five sending stations will use a different channel (frequency + offset) to send a single message to a receiving station. As they are using multiple channels, the messages can all be sent simultaneously. As long as the receiving station is able to receive multiple message transfers simultaneously, as is the case when using HRRM with the js8call opmode, the receiving station will receive these message as they are being sent. To summarize this, the total time on air for this transfer of the five messages using this approach is total time = 1 x message transfer time.

7.a.vii Scenario 2 Comparison

To compare these approaches, scenario 2b completes the message transfer twice as quickly as scenario 2a. Or to put this another way, scenario 2b can use a modulation mode that is half the speed of that used by 2a and still transfer the messages in the same amount of time.

Now compare 2a with 2c. Scenario 2c has the added benefit that it is able to utilize multiple channels simultaneously in addition to the other benefits of real-time simultaneous parallel communication. The net effect of scenario 2c compared to scenario 2a is that of a ten times multiplier; scenario 2c is 10x faster than scenario 2a. Or to put this another way, scenario 2c can use a modulation mode that is ten times slower than that used in scenario 2a and still transfer the messages in the same amount of time.

7.a.viii Comparison Summary

The above scenarios help to illustrate the degree to which the technique of real-time simultaneous parallel communication when used in a group setting is more efficient. In the examples given, when using a single channel the speed multiplier is between 2x and 6x times faster. When using multiple channels this speed multiplier is 10x times faster.

When these improvements are factored together with the HRRM optimizations of sending content only, dictionary compression and run length encoding, this linear performance improvement multiplier becomes exponential. This multiplier relates to aspects of communication other than the different digital modulation modes themselves.

This gives a backdrop where it is clear there is plenty of headroom for the operator to select the digital mode that is appropriate, whether it is JS8 slow mode or Olivia or PSK or something else and not have to be unduly concerned about any net-negative impacts to the message transfer speed. The often stated criticism that “it’s too slow, it’s like watching paint dry” clearly does not apply in this context.

In reality, there may be some additional effects that come into play, that may adjust these multipliers up or down by some amount but it is clear that by using HRRM in a group setting along with real-time simultaneous parallel communication techniques, transferring messages between the stations is extremely efficient. These multiplier effects can be used to real benefit and may help address the challenges related to transferring messages via gateway stations.

Having described the many benefits of real-time simultaneous parallel communication, the other question relating to this technique is the overall manner in which it is used. That is described in the next sections.

8 - Appendix 3 - Single Mode Digital vs Multi Mode Digital

HRRM can be used in conjunction with a digital net, either using the same digital mode as that of the message transfer (single mode) or using a different digital mode (mixed mode). Having a digital net running at the same time that forms are being transferred is a way to coordinate the message traffic and may enable a more efficient message transfer process.

8.a - Digital Net Setting

Real-time simultaneous parallel communication can be used in a net setting. The idea is that a digital net would provide the coordination at a high level and then the individual stations would then move to a different channel to do the actual message transfer then return back to the net. Alternatively all stations could remain on a single channel throughout the entire process and the net and the individual message transfers would proceed one after the other.

8.b - Free-for-all

The other approach to doing this is currently hypothetical. This would look something like the many FT8 stations going back and forth on one of their main frequencies such as 7.074. The idea here is that each individual station would manage its own activities with regard to message transfer...in essence a free-for-all approach. The many different message transfer modes available in HRRM make this type of approach feasible.

8.c - Single Mode Digital

An example of this would be running a JS8 net and using HRRM to send forms to the group over JS8.

One big advantage of using JS8 and JS8Call is that the narrow bandwidth of the JS8 modes allows for many stations to be present on frequency at the same time. This enables a single station to receive multiple messages from multiple stations at the same time. This multiplier effect increases message transfer efficiency significantly.

8.d - Mixed Mode or Multi-Mode Digital

Multimode digital requires the ability to use two different digital modes at the same time. When multiple digital modes are used it is possible to receive and decode the different modes of communication simultaneously. Only one of the modes is used to transmit at any given time.

One reason for doing multi mode digital is that some modes are better suited for running a net and other modes are better suited for passing message traffic. When running a net and passing message traffic on the same frequency, the use of two different modes may be desirable and also provides additional flexibility.

There are several approaches to doing multimode digital as follows:

1. Using two different fldigi modes in one application on one computer using one radio. One for running a net and one for sending message traffic
2. Using two different js8call modes in one application on one computer using one radio. One for running a net and one for sending message traffic
3. Running two separate programs such as fldigi and js8call side by side on the same computer using one radio. Js8call could be used for running the net and fldigi for passing message traffic.
4. Running two separate programs such as fldigi and js8call on two different computers using two radios.

Option 1 above can be achieved simply by typing directly into fldigi to run a net using a particular mode then when it is time to transfer messages, switch to HRRM and let HRRM handle the message transfer using the mode selected within HRRM.

Option 2 above can also be realized with the following. Running a js8 net using the js8-net program on github (https://github.com/gh42lb/js8-net). HRRM can be run alongside js8-net. The only accommodation that needs to be made is in the js8call external API settings to allow at least 2 ports to use the JSON interface instead of 1.

Option 3 above. Although this is experimental, I have confirmed that this is quite possible at least on the raspberry pi 4b. In order to use this approach there are several pre-requisites as follows:

1. Configure JS8Call to use the digital interface directly. I use a signalink and the JS8 audio settings are configured to point to the alsa_input.usb-BurrBrown_from_Texas_Instruments_USB_AUDIO_CODEC-00.analog-stereo and alsa_output.usb-BurrBrown_from_Texas_Instruments_USB_AUDIO_CODEC-00.analog-stereo devices.
2. Configure fldigi to use the default input and output.
3. Set the audio pop-up menu at the lower right of the raspberry pi screen to USB AUDIO CODEC for both input and output.
4. You must be using a VOX based setup either through a vox setting on your rig that detects audio signals and engages ptt accordingly or you can use an external device such as the Signalink for this purpose.

Using the above setup JS8Call and fldigi can be run at the same time and both applications can decode the respective traffic simultaneously in real time on the pi. Performance also appears to be acceptable.

Using this approach, multiple JS8 signals and a single fldigi signal can be received and decoded simultaneously. It would also be possible to use either JS8 or fldigi to transmit to other stations using a single digital mode such as fldigi for example to verify accurate message receipt. As soon as the message is transferred then both stations will continue to receive any transmissions for a net or other activity that may be going on either on JS8 or fldigi.

Option 4 above. This is also possible simply by running the net from one computer and running HRRM on another computer. Participation on the net can be done from the first computer. When it is time to transfer forms, the operator would make sure all transmits regarding the net are complete, then pause everything on the first computer, then switch to the second computer to transfer the message. Once message transfer and all transmits are complete, the operator would then switch back to the first computer to continue on the net.