Isode

babcock

UK MOD XMPP OVER HF PILOT

Steve Kille

CEO Isode Ltd

14th February 2019

Contents

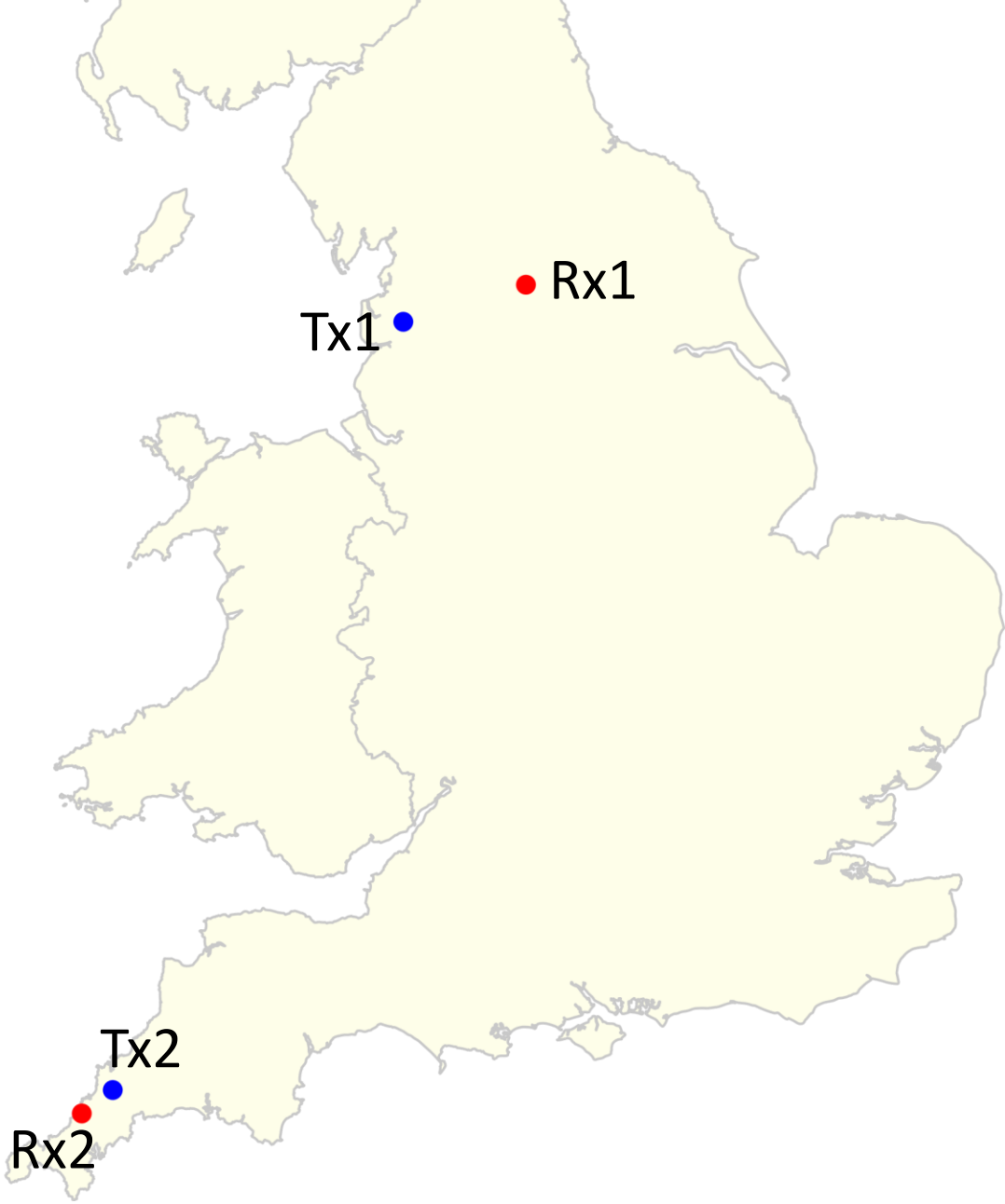
- Background and Goals
- Test Setup
- Capabilities Demonstrated
- Measurements
- What Next

Background and Goals

- UK MoD Funded Babcock to run an XMPP over HF trial using Isode XMPP Software
- Real Time Chat has become a mission critical service
  - XMPP is the NATO Standard (widely known as the JCHAT service)
  - “Group Chat” provided by XMPP Multi-User Chat (MUC) is the core service
- Highly desirable to use Real Time Chat for Naval and Airborne communication when HF is the only available bearer
- Trial run to evaluate viability of providing this service

HF Configuration

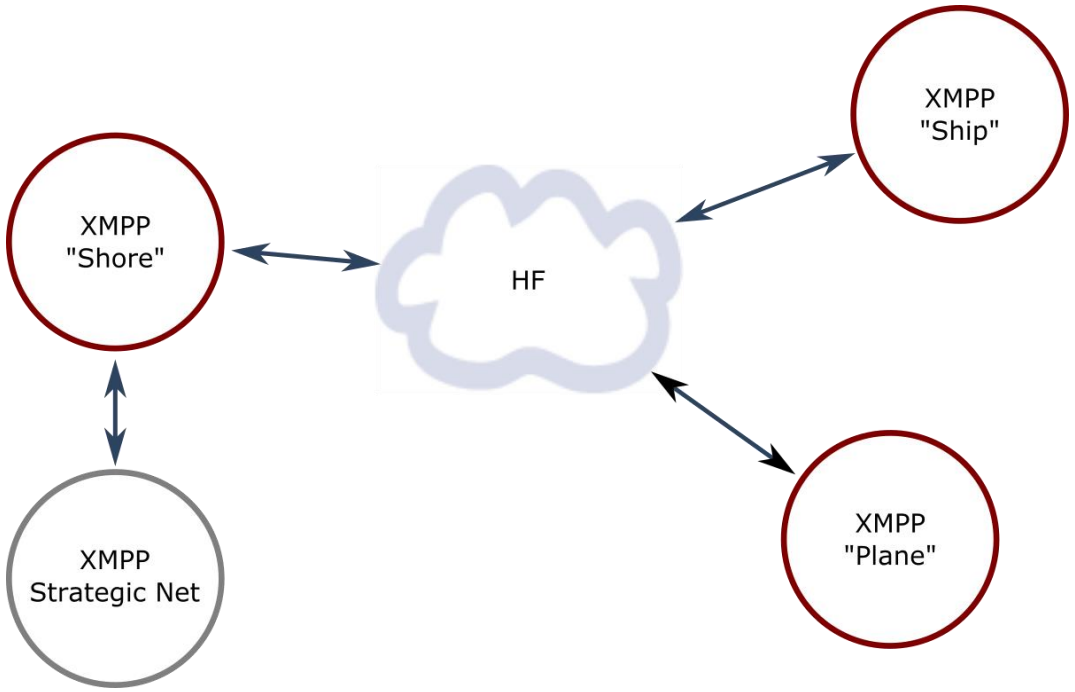
- DHFCS UK Middle: Separate Transmit (Tx1) & Receive (Rx1)
- DCHFS UK South: Separate Transmit (Tx2) & Receive (Rx2)
- Rockwell Collins Q9600 Modems, operated at selected fixed speeds
- Rockwell Collins RT2200 Radios
- Everything run from DHFCS UK Middle Receive site



Isode

XMPP Demo Setup

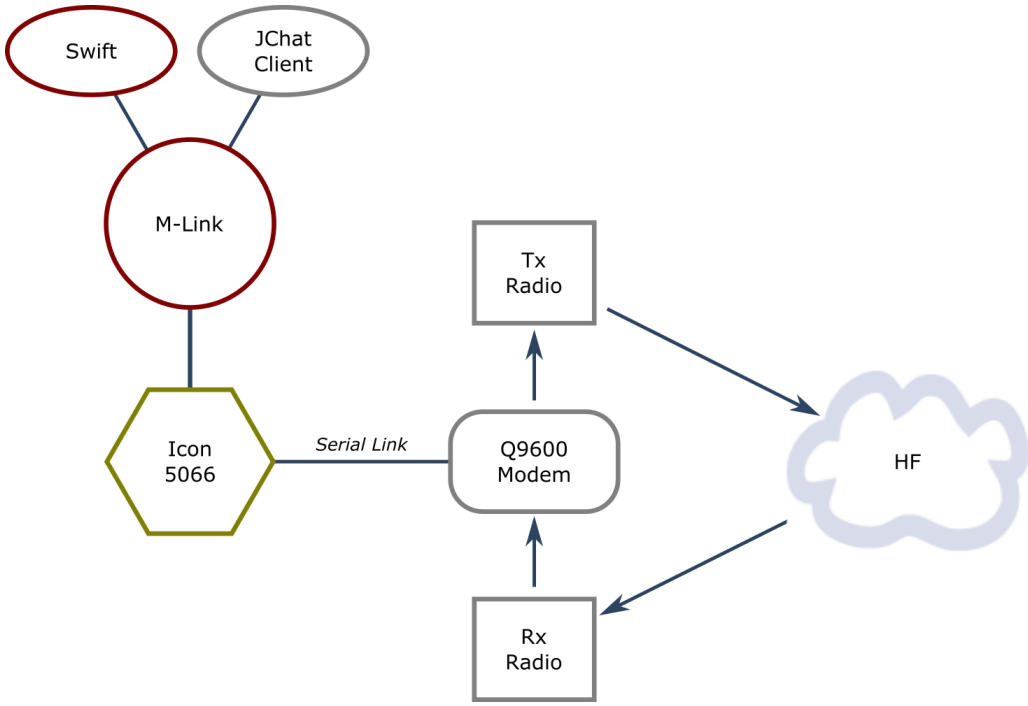
- "Plane" and "Ship" nodes
  - Both firmly on ground
  - Separate HF channels
  - Ship to Plane communication via shore
- XMPP Relay to fixed (strategic) network to enable users on network to communicate with Ship and Plane
  - Simulated by an extra XMPP server



Isode

Node Configuration

- Modem connected to Tx and Rx radios at separate sites
- Icon-5066
  - STANAG 5066 Server
  - Connected to modem over asynchronous serial
  - No modem control or SNR reporting
- M-Link
  - Isode XMPP Server
  - STANAG 5066 SIS connection to Icon-5066
- Two XMPP Clients
  - Swift (Isode product)
  - JChat (NATO Client)



Isode

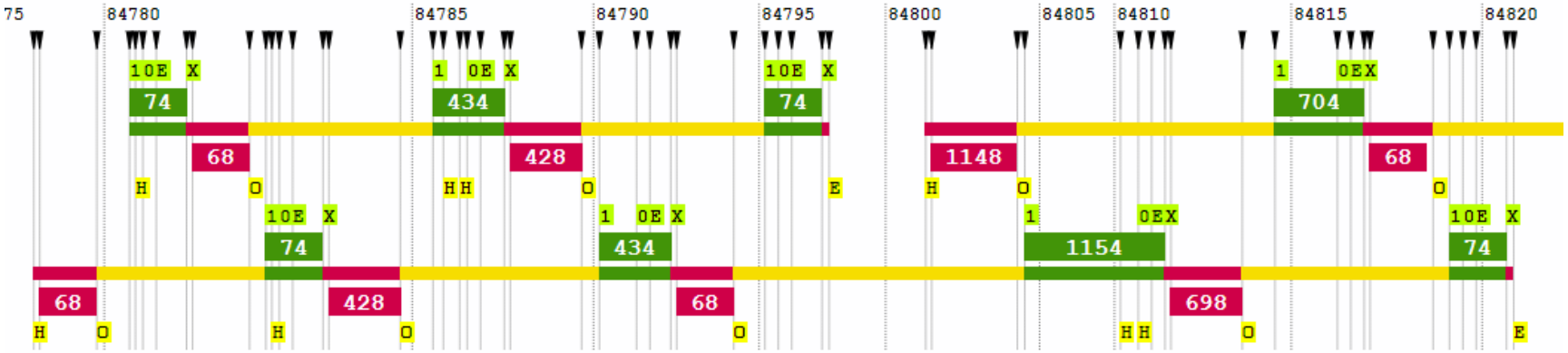
Core Capabilities Demonstrated

- Operation over STANAG 4539 waveform at fixed speed
  - Tests from 300bps to 4800bps
  - Periods of minutes and longer when no transmission possible
- User experience good – even at 300 bps
- Demonstrated
  - 1:1 Chat
  - MUC Rooms
  - File Transfer over XMPP
    - Not an efficient way to transfer files, so not generally recommended
    - Not interoperable between Swift and Jchat
      - Swift uses new JINGLE approach (XEP-0234)
- Comfortably passes the “it works nicely” user experience test
- Babcock Demo to MoD on 7th Feb 2019 at Forest Moor
  - Summary report: “Event went very well and the demo worked without issue”

Under the Hood

- A number of key technologies in M-Link enable this to happen
  - https://www.isode.com/products/m-link-constrained.html
  - Good performance dependent on capabilities not in open source XMPP servers
- XEP-0361 “Zero Handshake Server to Server protocol”
  - XMPP protocols have many handshakes on startup
  - This approach eliminates them
- XEP-0365 “Server to Server communication over STANAG 5066 ARQ”
  - Direct mapping onto STANAG 5066 RCOP
  - Eliminates TCP performance issues which arise if STANAG 5066 F.12 (IP Client) is used
- XEP-0289 “Federated MUC for Constrained Environments” (FMUC)
  - XMPP MUC is centralized on one server, which gives performance and reliability issues for HF
  - FMUC addresses this
- XMPP Trunking
  - https://www.isode.com/whitepapers/xmpp-trunking.html
  - Standard XMPP goes direct to final server, which will not support this sort of architecture
    - Strategic network connection is indirect; also plane/ship communication

OTA XMPP at 1200 bps



- This shows XMPP over STANAG 5066 at modem level
- Numbers along top at 5 second intervals
- Sender and receiver lines with byte counts
  - Red is transmit
  - Green is receive
- Can observe efficient mapping of application level down to modem layer

Isode

Beyond “it works nicely” user experience

- Good user experience is necessary, but insufficient
- Questions:
  - Can you measure how well it works?
  - Does it always work well?
  - Do messages ever get lost?
- XMPP Messages are generally small
  - Not bulk transfer, so throughput less important
  - Latency is the key measurement

Measuring XMPP Latency: Ex Latte

- Ex Latte is an Isode Tool for measuring XMPP Latency
  - Isode happy to share Ex Latte tool
- XMPP Client (bot) that runs on two or more nodes
- Generates messages at (optionally randomized) intervals
- Option for “probabilistic response”
  - Causes messages to group like chat messages do
- Messages contain:
  - Sequence Number
  - Timestamp
  - Text message randomly selected from list
- Supports 1:1 and MUC
  - Can watch the bots chatting in a MUC
- Runs create logs
- Ex Latte Analysis provides
  - Detection of message loss
    - Ideally zero
  - Latency Analysis

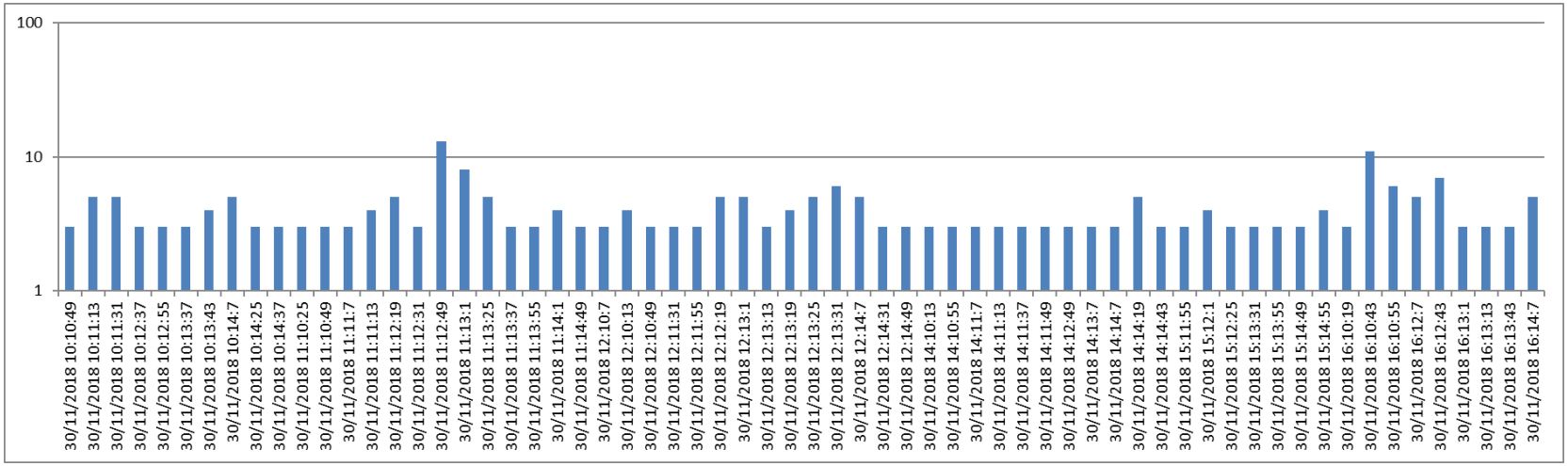
Measurements Made

- Babcock provided detailed report to MoD
- Test pattern sent messages randomly for 15 minutes, at 30 minute intervals
- Some runs were MUC and some 1:1
- Initial tests at 4800bps
  - 12 runs over several hours
  - The following graphs relate to these initial tests
- Official tests made at 600 bps
  - This was considered a safe speed (Babcock choice)
  - 13 Runs made over several hours each

Measurements: summary analysis

- Latency patterns from MUC and 1:1 were similar (indistinguishable)
- No message loss
- At 4800 bps
  - Most messages had 3 second latency at 4800 bps
    - This reflects immediate successful transmission
  - A lot of messages took 6-7 seconds at 4800 bps
    - Reflecting error on initial transmission and correction on ARQ retransmission
  - Some messages took longer – up to 15 minutes or so
- At 600bps
  - Base latency was 10 seconds
  - Patterns similar
  - A few very long outages with no transmission (one overnight)
    - Messages were all delivered in the morning
    - Comment “XMPP Works even when HF doesn’t”

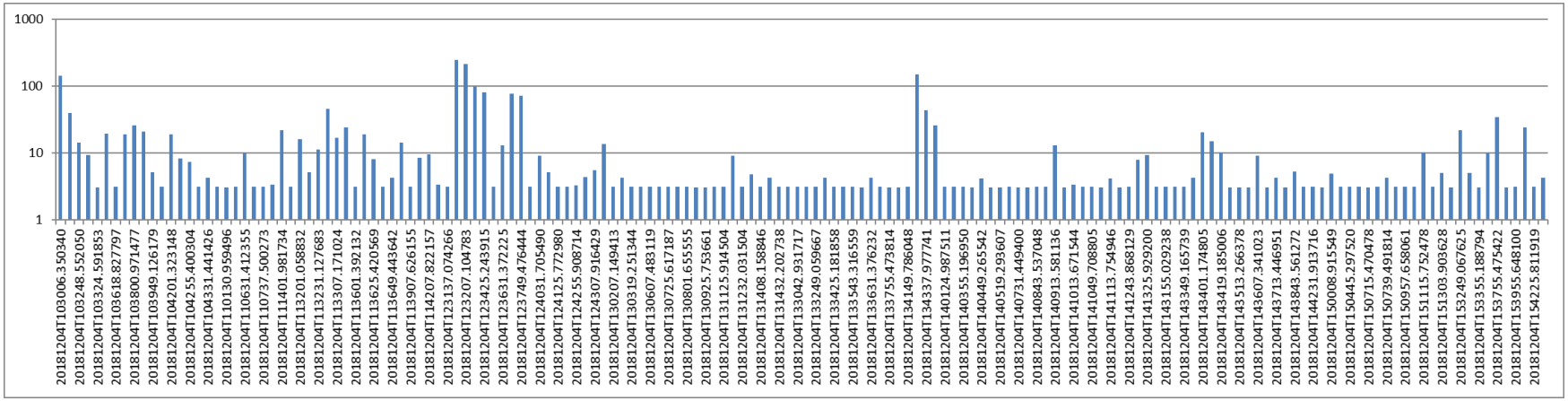
Run under good conditions



- 1:1 messages
- Most messages transferred in 3 seconds (immediate)
- Some messages retransmitted

Isode

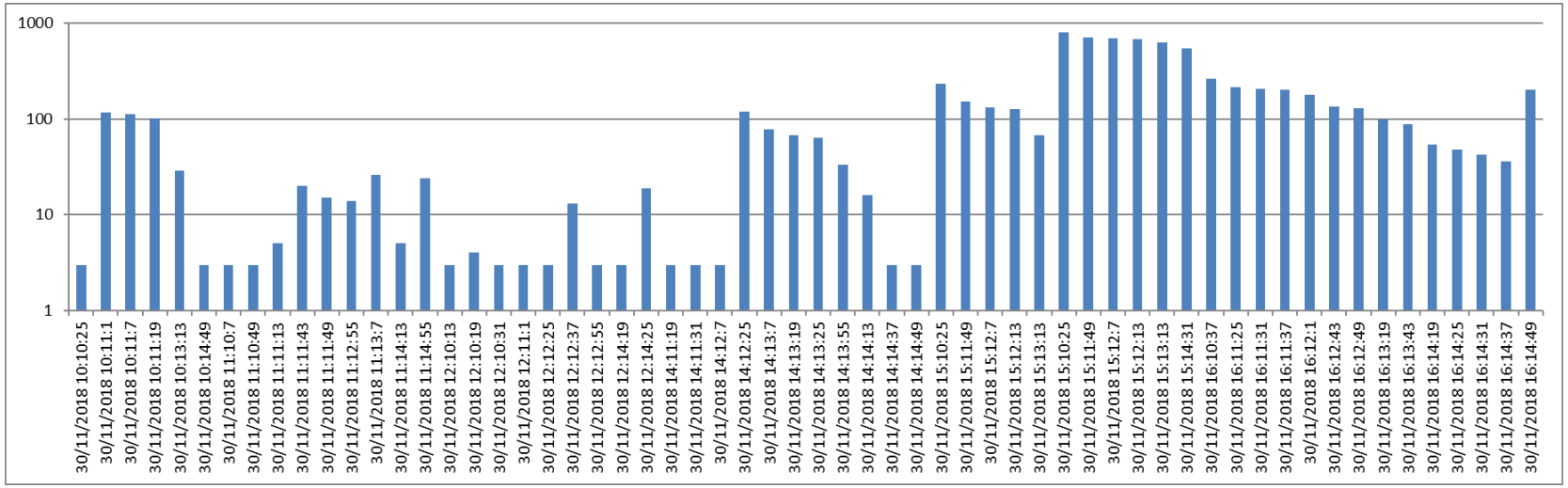
Run under less good conditions



- MUC messages
- Most messages transferred in 3 seconds (immediate)
- Some messages with delays up to two minutes or so
  - Receiving short transmission without EOTs led to 127.5 second (safe) wait
    - This explains most of the two minute delays
    - We have made some Icon-5066 product changes to avoid this delay

Isode

The Worst Run



- 1:1 messages
- Only two runs like this
- Quite a few messages take several minutes
- Slowest is 13 minutes
  - Logs suggest that rate change to lower transmission speed would help for some of these delays
    - Particularly where unidirectional transmission observed
  - In others, nothing getting through, so would need a new frequency (ALE)

Isode

What We Learned

- Narrowband HF (9600bps) is amply fast enough to support XMPP
- End of transmissions are not always detected by the modem
  - I talked about this at Bristol BLOS Comms meeting last year
- Monitoring SNR would have been really helpful
  - We speculate that delays on the bad runs could have been significantly improved by rate change to reduce transmission speed
  - Need more information as to what was going on

What Else Needs to be Tested?

To Test	Anticipated Result
Real Ships and Planes	Need to try and find out
Variable Speed	Improve latency performance
SNR Monitoring	Better understanding of what is happening when delays are unexpectedly high
4G ALE (Believe 2G ALE is too slow)	Better latency, due to selection of propagating frequency
Crypto	Moderate performance degradation
Operation in conjunction with bulk traffic	Moderate performance degradation
Channel sharing between nodes (STANAG 5066 Annex K or L)	Moderate performance degradation

Conclusions

- XMPP over Narrowband HF works nicely
- STANAG 5066 does not need anything special
  - Helpful to have full stack under Isode control
- XMPP Needs HF-oriented extensions
  - XEP-0361, XEP-0365, and XEP-0289 validated by trial
  - Open Source XMPP (e.g., Openfire) not appropriate
- Further trials are desirable

Questions?