Replacing 1200 Baud AFSK FM with PSK and Deploying DTN

Iain Young

g7iii@g7iii.net

Agenda

- Introductions
- Exactly What Are We Planning To Replace ?
- Why Replace It ?
- Computer Network Theory
- How We Replaced It
  - Tools and Utilities Needed
  - RF Setup and Maps
- AX.25 Link Established!
- Adding TCP/UDP/IP
- Adding DTN
- Results
- Conclusions
- Future Experiments
- Acknowledgements

Introductions

- Two major areas of focus
  - Replacing AX.25 1200 baud AFSK FM with PSK
  - Deploying DTN
- Who ?
  - Iain Young, G7III, MAXPAK Chairman
  - Dave Madew, M0DCM, MAXPAK Committee Member
- MAXPAK
  - Formed as a dedicated AX.25 Packet group for the Midlands
  - Constitution changed a couple of years ago, to include all digital modes

So What Are We Replacing, And What Are We Not Replacing ?

- The AX.25 Physical Layer
  - In “Network” Parlance, Layer 1
  - We mean the 1200 baud AFSK FM transmissions
- We are not replacing what the network world would call “Layer 2” or “The Data Link Layer” where you send and receive AX.25 frames to and from AX.25 addresses
- The mode we are all familiar with could be more formally described as AX.25 over 1200 baud AFSK FM
- The mode we will be creating could be more formally described as AX.25 over PSK

Why Do This ? (1)

- 1200 baud AFSK FM is not exactly well known for it's effeciency, especially with regards to:
  - Spectrum usage,
  - Power requirements
  - Link budget
- Even it's 300 baud cousin is not particularly well thought of on HF, can be improved on and that's only a quarter the throughput!

Why Do This ? (2)

- Plenty have claimed PSK is superior
  - but it's hard to find real figures
  - so can we prove it ?
- We can eliminate the dedicated hardware TNC
  - Most of us will have a soundcard interface of some type already
  - One of the issues with “packet” is the need for a dedicated TNC
- We can do better
- We need to do better!
- Fun!

ISO Layer Models

Pure ISO Model		
Layer 4 (Transport Layer)		
Layer 3 (Network Layer)		
Layer 2 (Data Link Layer)		
Layer 1 (Physical Layer)		

Typical Home Network		
TCP		UDP
Internet Protocol (IP)		
Ethernet		
Wifi, CAT5, CAT6 etc		

Typical AX.25 (+IP) Setup		
	TCP / UDP	
NETROM	IP	ROSE
AX.25 L2		
1200 Baud AFSK FM		300 Baud

AX25 over PSK with IP and DTN Setup		
DTN Bundle Layer		
AX.25CMCL	TCP-CL	UDP-CL
	TCP	UDP
	IP	
AX.25 L2		
PSK125,PSK250,PSK500...Any PSK		

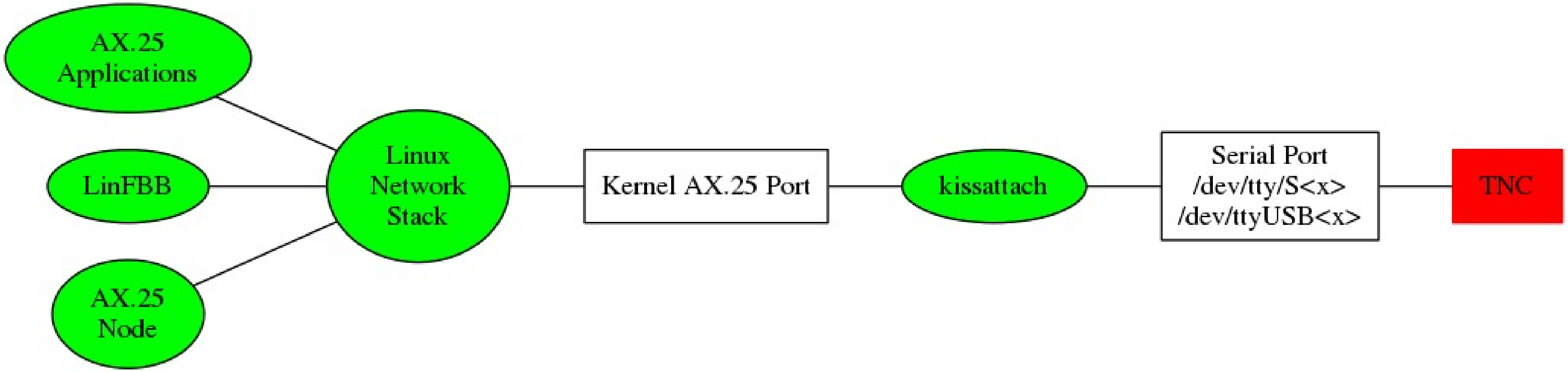
How We Did It

- A Beta version of fldigi can act as a software KISS TNC
- Since this is software:
  - A Soundcard interface, such as a SL-1 or SL-USB is needed
  - Linux AX.25 libraries and utilities
  - A utility was needed to link fldigi to the rest of the Linux AX.25 stack
    - A note here: Linux has AX.25 support in the kernel, just like ethernet and tcp/ip etc

socat

- A Linux utility. The name Stands for Socket CAT
- Forms a pipe between different types of sockets, files, serial ports, pseudo terminals etc
- Since KISS TNCs are essentially pseudo terminals under Linux, then one end can connect to a pseudo tty, while the other is connected to the appropriate UDP port for fldigi
- Allows us to run fldigi on the local machine, or a remote machine

Linux and AX.25 – Traditional TNC



AX.25 Applications

LinFBB

AX.25 Node

Linux Network Stack

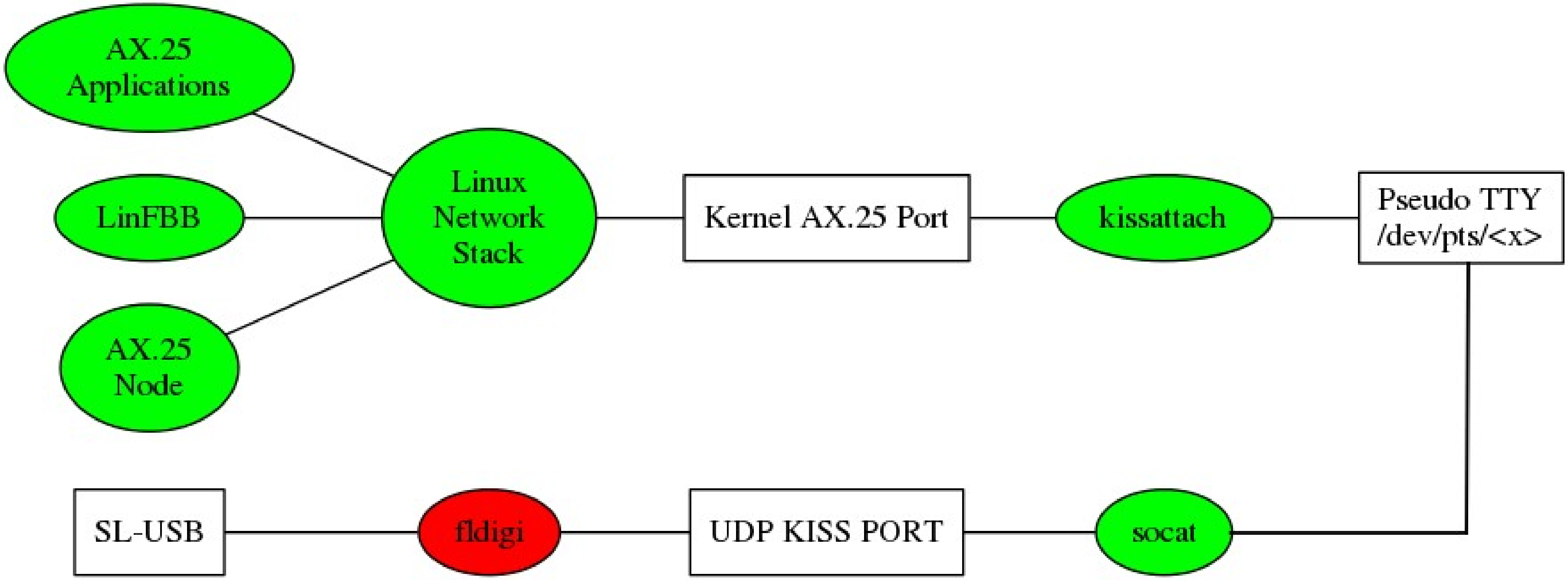
Kernel AX.25 Port

kissattach

Serial Port
/dev/tty/S<x>
/dev/ttyUSB<x>

TNC

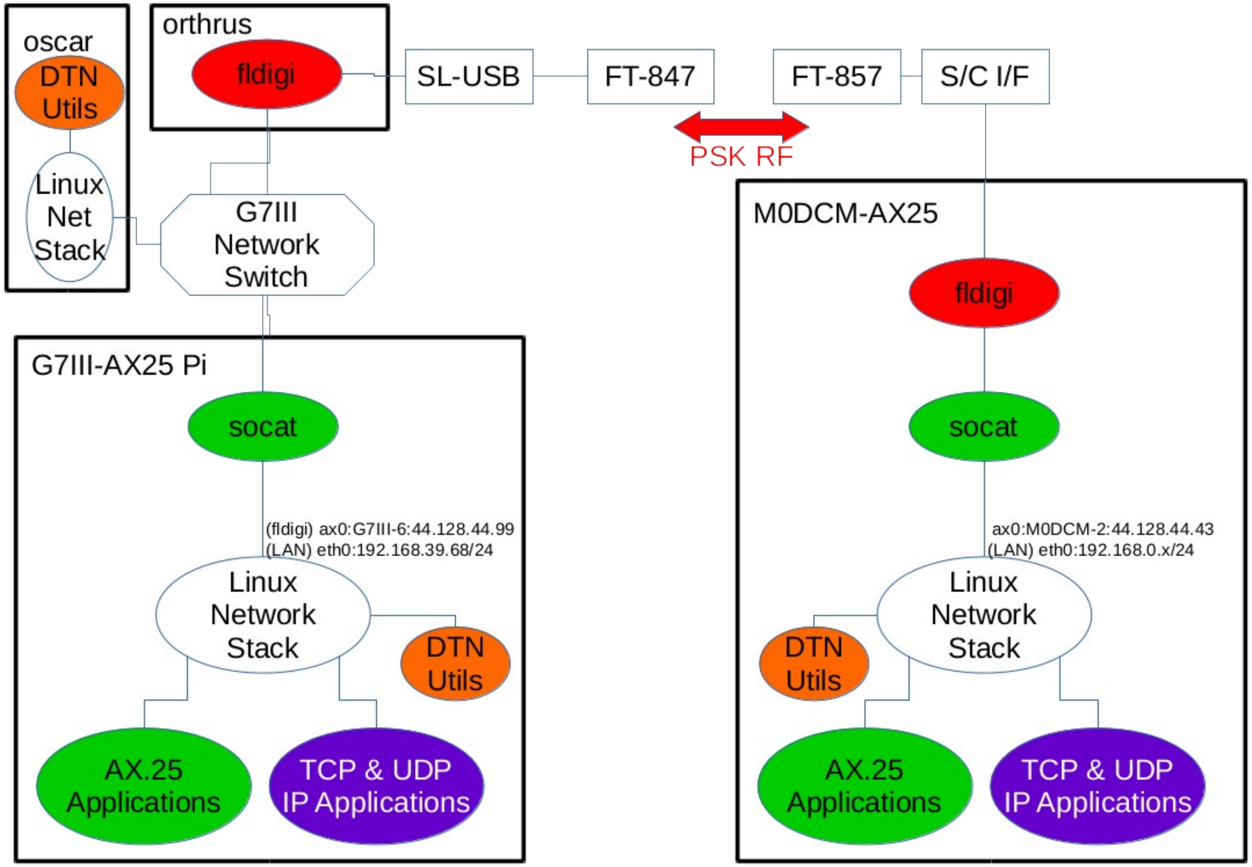
Linux and AX.25 – Software TNC - fldigi



Setup

- G7III
  - FT-847
  - 6 ele crossed Quad
  - Raspberry Pi running AX.25 stack
  - Intel x86 machine running fldigi
- M0DCM
  - FT-857
  - Vertical Co-Linear
  - Intel x86 laptop running both fldigi and AX.25 stack
- Initial tests were done at 50W to ensure a good link (as it is quite a difficult path between G7III and M0DCM), but then realised 5W would suffice quite by accident, with no loss of throughput!
- Further tests were carried out at 20W on 70cms

Network Diagram



So What Does This Setup Allow Us To Do ?

- Connect To Each Other Over AX.25
  - Eg C M0DCM-13
- Use all the Linux AX.25 facilities, applications etc (eg node, LinFBB, converse etc)
- But rather than 1200 baud FM, we connect using PSK125, 250, 500 or whatever we choose
- But using Linux we can go further

Adding TCP and UDP support

- We can add TCP/IP very easily.
  - Just add an appropriate IP address to the kernel AX.25 interface, and you have UDP/IP and TCP/IP avaliable
- This was done, and we were able to “ping” each node, and transfer UDP and TCP packets
  - We did have an issue with TCP and handshaking, that we never entirely got to the bottom of, but we think it was the length of time M0DCM's interface was taking to switch beween transmit and receive
- This gave us the ability to have a full IP stack – DNS, HTTP, NTP, SMTP, POP3 etc – Any standard IP protocol out there
- But we can go even further...And add DTN!

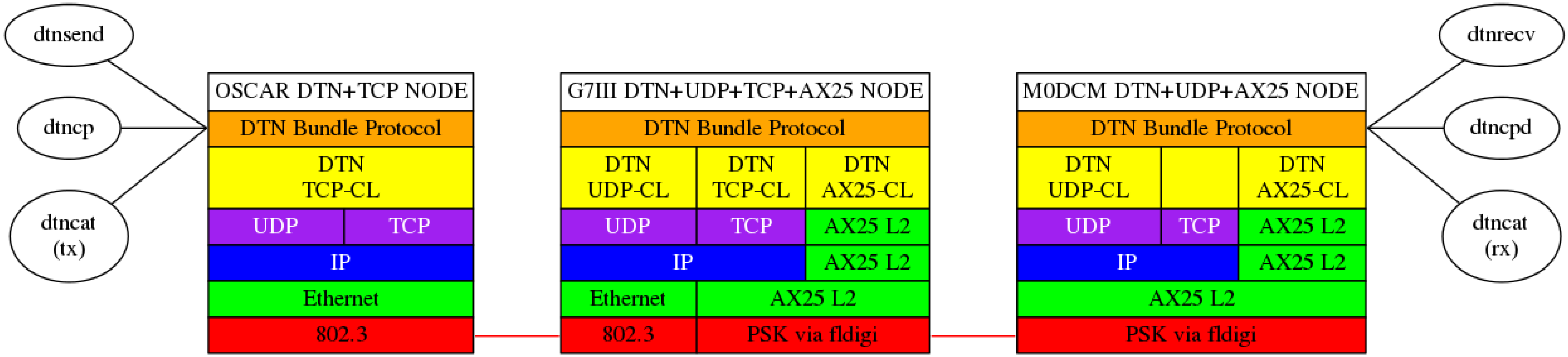
What is DTN ?

- DTN is “Delay|Disruptive Tolernant Networking”
  - Designed for noisy, unreliable, or non constant links
  - Consists of a “bundle” protocol which sits on top of a number of “convegence layers”, which are used to actually transmit the messages and files
  - Allows nodes with different convergence layers to communicate, as long as an intermediate node, or nodes have the needed convegence layers
  - Convergence Layers are typically Layer 2 (EG Ethernet, Serial) or Layer 3 or 4 (eg TCP, UDP)

DTN Reference Implementation

- Aka DTN2
- Has some “Interesting” Convergence Layers:
  - TCP
  - UDP
  - AX.25 Connected Mode
- We already have AX.25, and TCP/IP over AX.25 over PSK configured, so:
- We can set up a DTN network using Linux, AX.25, TCP, UDP, FLDigi, and a Soundcard Interface!

DTN Network Layers



dtnsend

dtncp

dtncat (tx)

OSCAR DTN+TCP NODE

DTN Bundle Protocol

DTN TCP-CL

UDP | TCP

IP

Ethernet

802.3

G7III DTN+UDP+TCP+AX25 NODE

DTN Bundle Protocol

DTN UDP-CL | DTN TCP-CL | DTN AX25-CL

UDP | TCP | AX25 L2

IP | AX25 L2

Ethernet | AX25 L2

802.3 | PSK via fldigi

M0DCM DTN+UDP+AX25 NODE

DTN Bundle Protocol

DTN UDP-CL | | DTN AX25-CL

UDP | TCP | AX25 L2

IP | AX25 L2

AX25 L2

PSK via fldigi

dtnrecv

dtncpd

dtncat (rx)

- “oscar” only has DTN's TCP-CL. M0DCM only has DTN's AX25CM-CL and UDP-CL
- But G7III has all three. Thus oscar and M0DCM can exchange data via DTN
- Could have done DTN over Ethernet between oscar and G7III
- (Note all three nodes have all DTN processes, it was just impossible to fit them all on one slide easily)

Testing Objectives

- Prove that we can replace AX.25 1200 AFSK with PSK
- Prove that we can run DTN over AX.25, over TCP/AX.25, and UDP/AX.25
- Measure the throughput when using PSK rather than FM
- Measure the improved throughput when using multi carrier PSK modes
- Measure the effect of changing the MTU / Packet Size
- And to have fun!

Testing Method

- Check F3E (FM Voice) and J3E (SSB Voice) communications
  - Both stations heard each other
  - But even at 50 Watts, neither had great signals, and certainly not S9!
- Check L1 and L2 Connectivity
  - Waterfall during remote station transmission
  - Confirm AX.25 beacons were correctly decoded
- Check L3 IP Connectivity
  - Ping between the two stations
  - UDP test via nc (network commander)
- Transfer Tests
  - Using DTN
  - Measure throughput of many PSK Modes

Measurement Method

- Looking at the raw RF Transmission Time
- Not The DTN Reported Time
  - AX.25 Stack Delays
  - Up to 20 seconds!

What Were We Able To Achieve ? (1)

- Setup a L2 AX.25 network between M0DCM and G7III using PSK as a Physical Layer
- Add TCP/IP over that, and use IP Protocols (TCP and UDP) to transfer information
- Run NTP over UDP/AX.25
  - Just for kicks - it worked, but that's an entirely different presentation!
- Add DTN over UDP and (later) over AX.25
- Transfer files and messages between the two nodes via DTN/UDP, and DTN/AX.25

What Were We Able To Achieve ? (2)

- Use different PSK modes
  - PSK125, PSK250, PSK500, PSK500C2, PSK800C2 were all successfully tested
- Add a third DTN node at G7III (“oscar”), with only DTN/TCP support
  - And then send DTN messages and files from to and from oscar and M0DCM, via G7III acting as a “protocol translator”
  - DTN Messages and files were sent via TCP from oscar to G7III, and from G7III to M0DCM over AX.25 or UDP, before ending up at M0DCM's DTN endpoint.
- Run Transfer Tests between the two AX25/DTN nodes

Some Actual Transfer Results

G7III / M0DCM DTN/UDP/AX25UI/PSK Throughput Tests										
	BW	240 bytes		512 bytes		1k bytes		1.5k bytes		
	Hz	Time	B/Sec	Time	B/Sec	Time	B/Sec	Time	B/Sec	
DTN/UDP/AX25/PSK125	125	31	7.74	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	
DTN/UDP/AX25/PSK250	250	16	15.00	27	18.96	FAIL	FAIL	FAIL	FAIL	
DTN/UDP/AX25/PSK500	500	9	26.67	17	30.12	25	40.96	34	45.18	
PSK500C2	1K0	6	40.00	8	64.00	14	73.14	20	76.80	
PSK800C2 MTU 255	1K9	2	120.0	N/R	N/R	N/R	N/R	N/R	N/R	
PSK800C2 MTU 576	1K9	2	120.0	4	128.0	N/R	N/R	N/R	N/R	
PSK800C2 MTU 1500	1K9	2	120.0	4	128.0	8	128.0	N/R	N/R	

- Failures were due to UDP Fragmentation Re-assembly timeouts
- Should be able to be fixed by:
  - Adjusting the MTU / Packetsize to align UDP and AX25
  - Modifiying the UDP Timeouts

Some Actual Transfer Results

G7III / M0DCM DTN/UDP/AX25UI/PSK Throughput Tests									
	BW	1 kilobyte		2 kilobytes		4 kilobytes		5 kilobytes	
	Hz	Time	B/Sec	Time	B/Sec	Time	B/Sec	Time	B/Sec
PSK500C2	1K0	14	73.1	22	93.1	FAIL	FAIL	FAIL	FAIL
PSK800C2 MTU 255	1K9			14	146.3	FAIL	FAIL	FAIL	FAIL
PSK800C2 MTU 576	1K9			14	146.3	24	170.7	29	176.6
PSK800C2 MTU 1500	1K9	8	128.0	14	146.3	22	186.2	28	182.9

- Failures were again, due to UDP Fragmentation Re-assembly timeouts
- But note we can transfer files of much larger size before it becomes a problem

AX.25 1200 Baud AFSK FM Bandwidth

- Varies depending on your deviation, but supposedly:
  - 2.5 kHz for 12.5 kHz channel spacing
  - 5 kHz for 25 kHz channel spacing
  - Have seen one implementation that claimed 3.4kHz...
    - Same source claimed 900Hz for HF AX.25
  - But these are figures are typically measured as +/-...
  - Anyone know a typical deviation figure for an Amateur Radio s/c that transmits 1200 baud AFSK ?

Some Context and Comparisons

EI7IG / G0HWW (FM) vs G7III / M0DCM (PSK) Throughput Comparison				
	B/W	Time (secs)	File Size	Bytes/Sec
Raw AX25CM/1200 AFSK FM	5k0	77	9177	119.2
DTN/AX25CM/1200 AFSK FM	5k0	108.33	9177	84.7
DTN/UDP/AX25UI/PSK500C2	1k0	22	2048	93.1
DTN/UDP/AX25UI/PSK800C2 MTU 255	1k9	14	2048	146.3
DTN/UDP/AX25UI/PSK800C2 MTU 576	1k9	24	4096	170.7
DTN/UDP/AX25UI/PSK800C2 MTU 1500	1k9	22	4096	186.2

EI7IG / G0HWW (FM) vs G7III / M0DCM (PSK) Throughput Comparison				
	B/W	Time (secs)	File Size	Bytes/Sec
Raw AX25/1200 AFSK FM	5k0	77	9177	119.18
DTN/AX25/1200 AFSK FM	5k0	108.33	9177	84.72
DTN//AX25CM/PSK800C2 MTU 1500	1k9	188	9177	48.81

Conclusions (AX25 vs PSK)

- AX.25 over PSK:
  - Can be done
  - Can get better throughput for less bandwidth than it's FM cousin
  - Can work over a noiser channel than AX.25 over FM
  - Can work over much less bandwidth, but at the trade off of throughput and efficiency, so HF ops a possibility
- However:
  - You must get packet lengths, MTUs, Time-outs between different network layers aligned correctly for optimal throughput

Conclusions (DTN)

- DTN is deployable over Amateur Radio Networks Today
- However, this is very much the low level protocols. Further work would be needed to integrate it into things like FBB, SMTP Mail etc
- That said, some of the work (eg to integrate SMTP with DTN using dtncat) would be fairly simple with certain MTAs (Postfix comes to mind)

What Didn't Work So Well ? (1)

- TCP over AX.25 UI as a convergence layer
  - This was put down to M0DCM's soundcard interface relying on a serial line to switch between RX and TX, and taking too long to switch back to RX
  - Where as G7III's SL-1 was using VOX
  - Situation was slightly improved by fiddling with some traffic shaping options on G7III's side to delay the KISS packets into fldigi
- AX25 Connected Mode as a convergence layer
  - Similiar issues to TCP over AX.25 UI
  - Finally worked with multi carrier (PSK800C2) and increased MTU (1500)

What Didn't Work So Well ? (2)

- We could not get PSK 1000 to work, even with higher power. Suspisions were:
  - Frequency Accuracy/Carrier recovery issues
  - HPF filtering affect of G7III's SL-1 interface
  - Yet we could get dual carrier PSK 500 and 800 to work!

Future Experiments

- Fix MTU/Packet Length mismatches and time out issues between UDP and AX.25
- Different Convergence Layers (NORM-CL)
- Different Modes, More Nodes, Higher Speeds
- Adding network latency on the LAN or radio interfaces to simulate orbital or even inter-planetary distances
- USRP / SDR Experiments for even higher speeds
- Spacecraft Links rather than just terrestrial simplex
  - P4A ?
  - Latency, Throughput, Double-hop Tests etc
- An Ad-hoc network rather than a fixed network like the old packet network ended up being ?

Future Experiments

- So anyone that:
  - Is within RF range of G7III (Coventry), and/or M0DCM (Walsall) on 2/70cms
  - Has a Linux box (A Raspberry Pi or Beaglebone!)
  - Has a Soundcard Interface such as a SL1 or similar

- Please get in touch!
  - Installation instructions for the beta version of fldigi, the DTN suite, and the Linux AX.25 stack can be provided, but prior Linux knowledge would be of benefit!
  - If we add just one more station, we can then have people within RF range of them, and so on and so forth!

Future Experiments

- USRP / SDR Possibilities
  - USRP can provide far higher symbol rates than fldigi
  - Even a RX only SDR could receive bundles via UDP. Acknowledgements could be sent back over a slower fldigi link or even internet if we had to!
- Consider integrating freedv, or CODEC2 etc
    - A digi-modes/DTN SDR groundstation ready for P4A ?
    - Just imagine what symbol rate you could get with 250kHz on the 8MHz transponder!

A Plea To Satellite Designers

- Please consider replacing AX.25 FM with PSK, even if you use AX.25 Frames, As we have seen, it is far more spectrally efficient
- It is even beneficial on link budget, as it requires less S/N ratio (although we have not quantified it in this presentation)
- You can trade throughput (symbol rate) for link and/or power budget if you need to.
- Software is plentiful today for decoding PSK, removing the need for a dedicated TNC

Acknowledgments

- Dave Maydew (M0DCM)
  - For being the “B” end of the link
- Darren Long (G0HWW) / John Ronan (EI7IG)
  - For the AX.25 Connected Mode DTN Convergence Layer
  - For various papers and previous tests comparing pure AX.25 vs DTN/AX.25
- Fldigi Team
  - For the Fldigi beta with KISS support
  - Without it, this would have been much harder