The SDSL Journey

Reverse engineering, open source connectivity and ARPANET replacement

Michael Sokolov

Harhan Engineering Co.
Open WAN Connectivity Project

What is SDSL?

- Same *low* speed up and down
- Pay more for *less* downstream bandwidth!
- ‘‘Business service’’: it’s all about the feeling of being **elite**
- Can be used as ARPANET replacement

Who offers SDSL?

USA-specific

Back-end operators:

- Rhythms/WorldCom (gone)
- Covad
- DSL.net (now MegaPath)

Front-end providers:

- AT&T
- Covad.net
- MegaPath
- Many small ISPs going through Covad

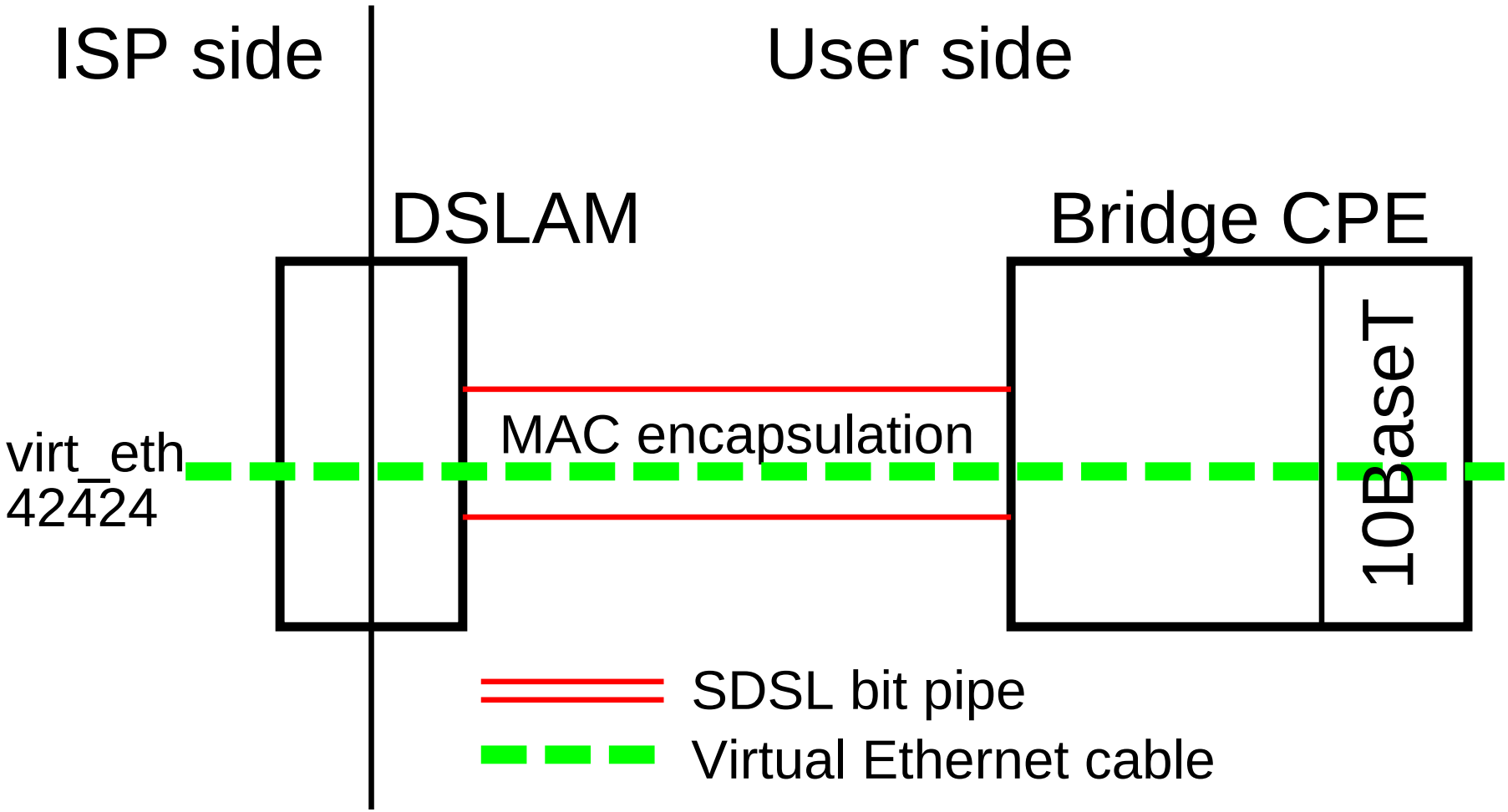
Bridging vs. true routing

SDSL does *NOT* use PPPoE!

Networking models used with SDSL:

- Bridging / MER
- True routed circuit
- PPPoA (Covad.net only)

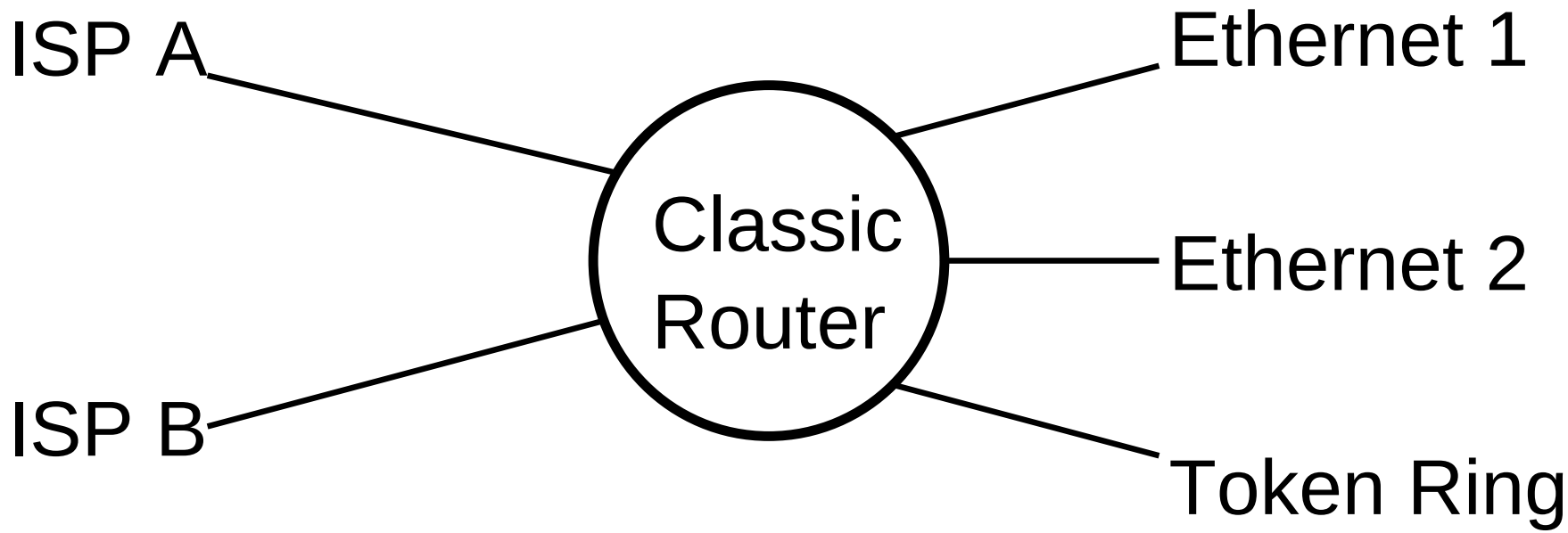
Bridged SDSL circuit



Standard bridge CPE (Copper Mountain)

- CM flavor only, bridged circuits only
- Configurationless
- Converts between Ethernet-over-SDSL and real Ethernet

... but some of us prefer the routed network model:



Key difference in circuit config:

Bridged/MER circuit:

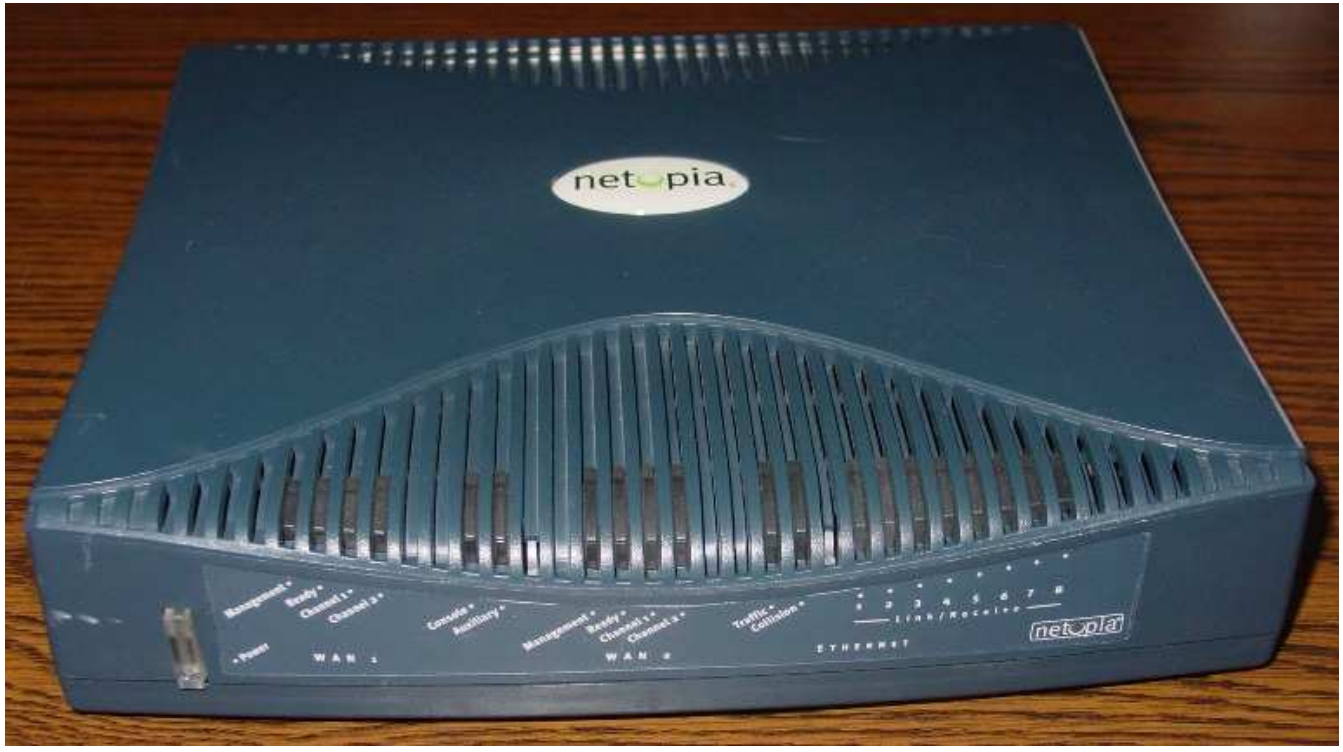
<dest MAC addr><src MAC addr><0806><ARP packet>
<dest MAC addr><src MAC addr><0800><IP packet>

True routed circuit:

<FR/ATM header><IP packet>

No MAC addresses and no ARP!

Standard CPE for routed circuits (circa 2001)

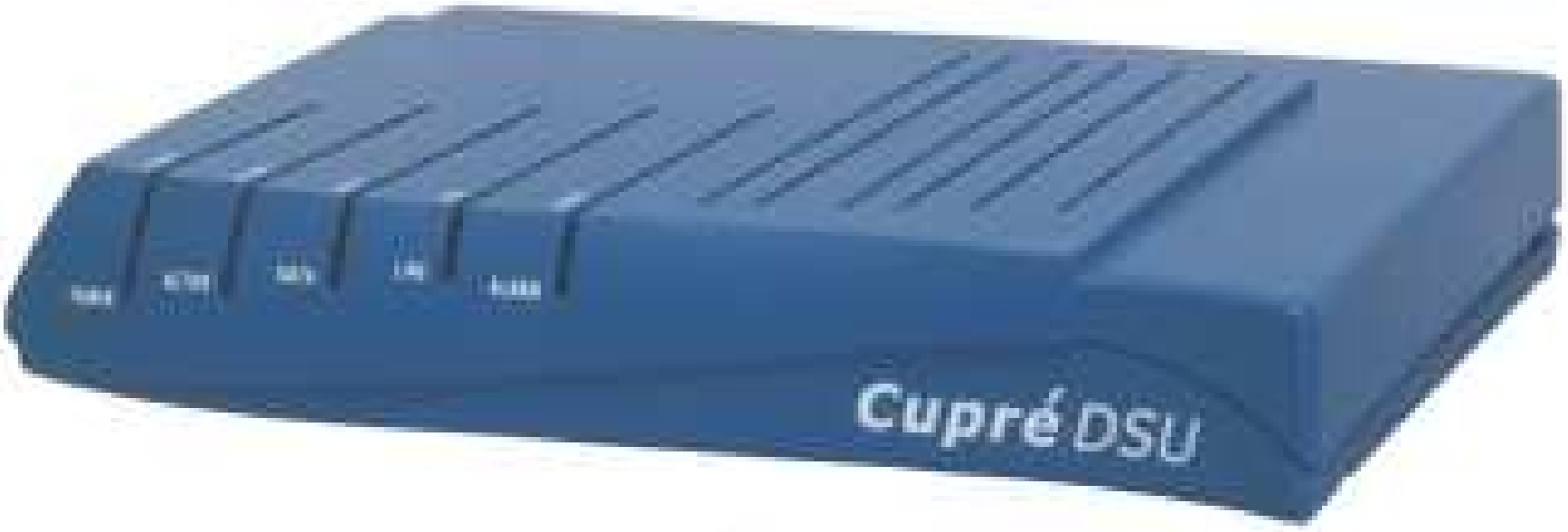


Yes, it’s a router, but it isn’t chosen by the end site’s own god!

Would like to extract the SDSL PHY part of those routers and offer it by itself, letting the user choose his own router. How do we do it?

- Social engineering
- Reverse engineering
- Confirmed the use of a standard HDLC bit stream (CM SDSL)
- Needed to identify the part of SDSL CPE called the ‘‘bitpump’’

Someone had already done this before us:



... but it was made of unobtainium, so I set out to recreate it.

What is SDSL in technical terms?

North American SDSL/2B1Q is *NOT* the same as ETSI SDSL or G.shdsl!

G.991 and ETSI standards are free and open, but useless for SDSL/2B1Q

What is SDSL/2B1Q in technical terms?

- De facto pseudostandard defined by the makers of Bt8960, Bt8970 and RS8973 chips
- Based on HDSL, an internal telco technology
- 2B1Q line code dates back to ISDN BRI

More generally:

- Full-duplex synchronous serial bit stream
- 2-wire transmission via echo-cancelling hybrid
- Choice of several data rates

What does it take to build SDSL CPE?

- Understanding the physics involved
- Using the RS8973 bitpump chip

- Software/firmware to control that chip
- SDSL flavors: each DSLAM vendor invented their own!

Flavors
Flavors
Flavors

Common invariant:

- Bitpump chip
- Sync serial bit stream
- 2B1Q line code
- HDSL heritage

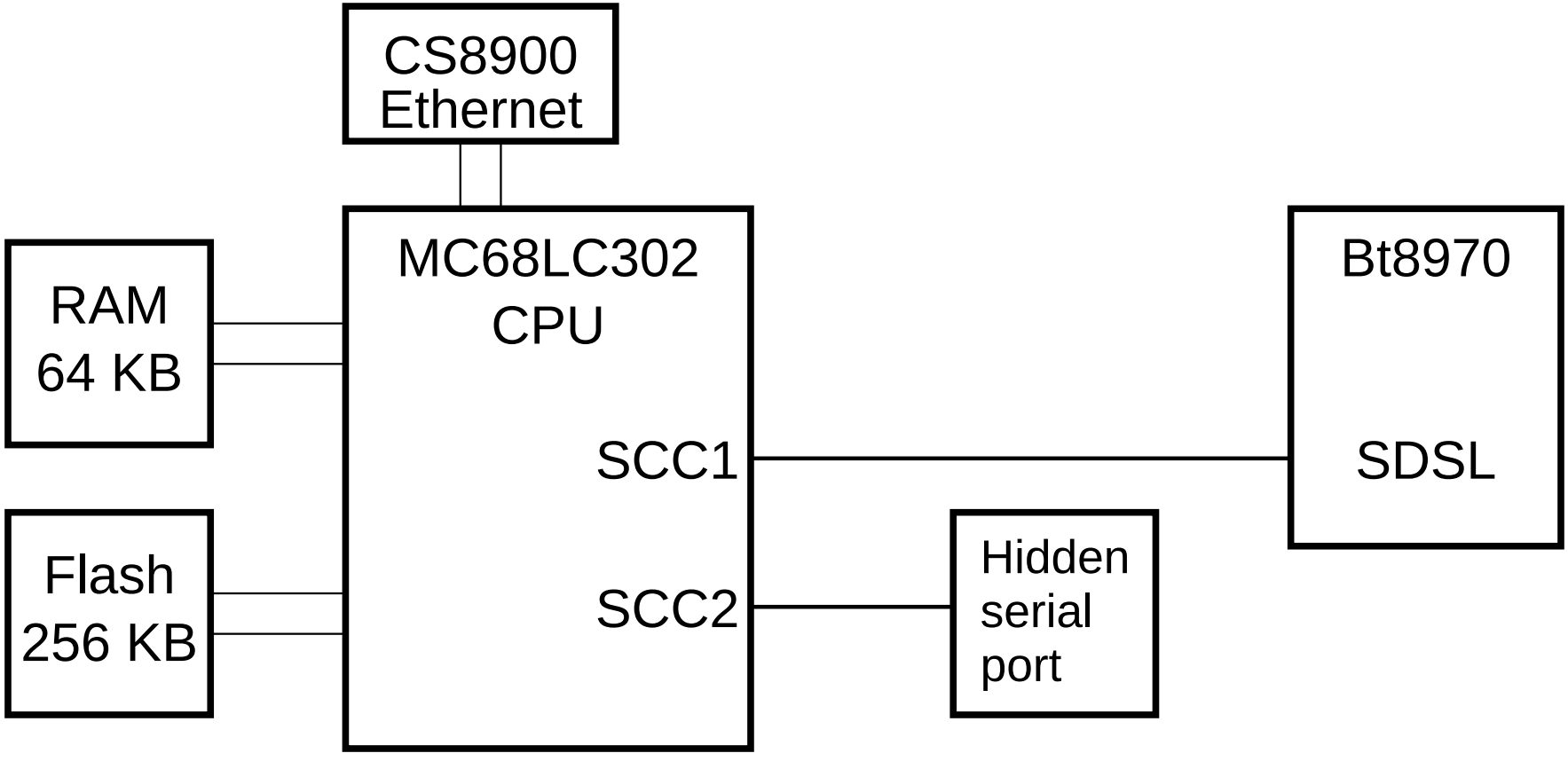
Flavor-dependent:

- Choice of data rates
- Quat orientation
- Pre-activation signaling
- Bit stream format

Strategies for open source connectivity to SDSL

- Bit-transparent DSU like the elusive Cupré box: great for Copper Mountain
- Layer 2 converter for SDSL/ATM
- Do both with a single hardware platform: OSDCU

Hack-o-Rocket: let's take an existing SDSL CPE board and turn it into a hacking instrument



Socketed PLCC32 flash chips were easy to read out and the M68K code was easy to disassemble!

Learned from CR201 firmware disassembly:

CM flavor of SDSL:

- Confirmed that it’s standard HDLC
- Speed autodetection reverse-engineered

Using CR201 hardware:

- Supplemented the data from physical H/W examination

By running our own code on CR201s hardware, we were able to:

- Play with the bitpump and put it through startup sequences
- Lay the software foundation for our own OSDCU
- Use the SCC to study the bit stream formats

Let’s look at the Nokia SDSL/ATM flavor now

Most of the Nokia reverse eng work was done in the absence of a real Covad line to test on!

- Beige box test: no pre-activation present
- Oscilloscope probe on QCLK: got the true data rates
- Run Hack-o-Rocket as HTU-C, linking up with Nokia flavor CPE
- Bit stream capture revealed the framing format

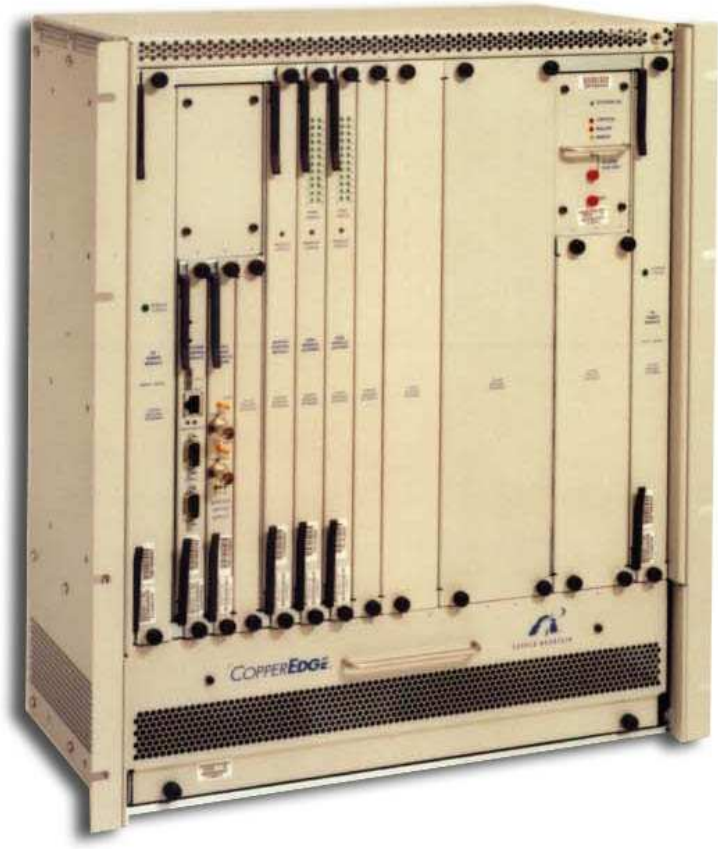
Proof of concept open source SDSL and IDSL implementation on the Hack-o-Rocket

- Tested and proven working on real Covad lines, both SDSL and IDSL.
- Demo given to a Covad install technician!

Finally built our own hardware!

SDSL in, EIA-530 out

Testing: CM DSLAM



Works like a charm!

More testing

- Verified interoperability with XSB-2000 DSUs
- Linking up with Netopia CPE (pretending to be a Nokia DSLAM)

Where do we go from here?

- Nokia DSLAM bring-up
- FPGA acceleration for Nokia L2 converter
- Bring the OSDCU board to production quality

http://ifctfvax.Harhan.ORG/OpenWAN/