THE TERASTREAM IPv6 NATIVE NETWORK ARCHITECTURE
How to build a modern service provider using IPv6 and Optics
Ian Farrer, Deutsche Telekom AG
TeraStream Motivations

Must address massive IP traffic growth driven by broadband access and new Internet services and Internet business models

Many networks and technologies, complex systems – long service lead-times, high-cost evolution to converged network architecture

Competitors offer better performance, more service flexibility and more features, faster provisioning, lower price

Multi-layer system complexity results in slow or lack of service innovation, low customer satisfaction, impacting revenue
One truly converged network - de-layered, IP and Optical are one, bits over wavelengths, digital over analog

The same technology for LAN and WAN - for LAN: IP packets in Home, Office, Data Center; for WAN: IP packets in Metro, Country, Continent

Digital services for consumers and businesses - communication, information, cloud-compute and -storage

Real-time OSS - instantaneous service provisioning, guaranteed good user experience

Cloud-era economics - service flexibility, fast-paced innovation, agile implementation, reduced system complexity, lower cost

Stay with the IP Internet Architecture using IPv6 for all functions and services
Have a standardized toolbox, “services in the data center”

Standard computers instead of specialized appliances
Look forward, what can we do, not backwards what we used to do

No L1, L2, L3 dependencies
No “best effort” you get what you pay for (Best effort is just a service class..)
## TeraStream Design Principles

<table>
<thead>
<tr>
<th>Principle</th>
<th>Applied to TeraStream design</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reduce the amount of technologies used</td>
<td>Use IP and optical transmission only</td>
</tr>
<tr>
<td></td>
<td>No OTN, L2, MPLS switching</td>
</tr>
<tr>
<td>Use IPv6 for all internal functions and services</td>
<td>No native IPv4 support in the network</td>
</tr>
<tr>
<td></td>
<td>IPv4 is a service</td>
</tr>
<tr>
<td></td>
<td>IPv6 based “carrier Ethernet service”</td>
</tr>
<tr>
<td>Avoid internal interfaces</td>
<td>Minimize non-customer, non-peering facing interfaces</td>
</tr>
<tr>
<td></td>
<td>Distribute Internet peerings, offload external traffic ASAP</td>
</tr>
<tr>
<td>Size the network to handle all IP traffic without IP packets losses</td>
<td>Dimension the network for peak hour IP traffic, no oversubscription, packet loss is extreme exception</td>
</tr>
<tr>
<td>Integrate optical networks and IP networks as much as possible</td>
<td>Integrate IP and optical layers into routers to simplify the network, avoid redundant mechanisms e.g. failure handling, reduce total cost</td>
</tr>
<tr>
<td>Use one network for all services – Internet, IP TV, business, ...</td>
<td>Single converged packet network</td>
</tr>
<tr>
<td></td>
<td>Note: Dominant traffic drives the design!</td>
</tr>
<tr>
<td>Deterministic and short routing path for all on-net traffic</td>
<td>Network distance between R1 access routers is at most two R2 backbone routers away and R1 is multi homed to two R2</td>
</tr>
<tr>
<td>Service policy for packets are outside the payload</td>
<td>Encode service type, traffic class, direction etc in the IPv6 address</td>
</tr>
<tr>
<td>Data Centers are directly connected to backbone routers</td>
<td>DCs connect directly to R2s to avoid building internal IP interfaces for very large amount of traffic</td>
</tr>
</tbody>
</table>
TeraStream – Design in a Nutshell

### TeraStream key functional elements

**R1**
- Terminate access interfaces
- Runs IPv6 routing only, integrates optical
- Access services
  - IPv6 - dealt with natively
  - IPv4 – IPv4 over IPv6 softwire between HGW / CPE and DC, R1 not involved
  - non-IP - L2-over-IPv6 encapsulation
- User configuration
  - using Netconf / Yang
  - Driven by real-time OSS i.e. self-service portal

**R2**
- Connects R1s, Data Centers and Internet peerings
- Runs IPv6 and IPv4 routing, integrates optical
- Closely integrated with Data Centers
  - Optimized handling of locally sourced services
- High scale IP bandwidth

**Data Center / Services**
- Distributed design
  - fully virtualized x86 compute and storage environment
- Network support functions - DNS, DHCP, NMS
- Real-time OSS incl. user self-service portal
- Cloud DC applications, XaaS services
- Complex network services e.g. high-touch subscriber handling
R1 ↔ R2: OPTICAL FIBER LINKS

80/96 WDM channels
100G Coherent DWDM Interop and Pluggable Technology

- Agree on a common set of parameters for the 100G line side
- Enable innovation by many players in the silicon optics arena
- Hard FEC, typ 800km
- If price is right, use in data center

- Coding
- Carrier Recovery
- Acquisition (blind)
- Reach
- Framing (works with both OTU4.4 and OTU4.10)
- Forward Error Correction (Hard FEC Staircase)
TeraStream user facing router R1

Service Guarantee Architecture

Configure per queue:
- Delay
- Drop
- Bandwidth
- Reorder
- Etc...

- IP traffic shaped to capabilities of L2 device
- 5000 customers connections per R1
- 20 * 10GE port for L2 device
- 4 * 100GE for R2 link
R2 router and traffic patterns

Traffic flow patterns:
- R1 ↔ Peers and Other R2 going north ↔ south (example: IPv6 Internet traffic)
- R1 ↔ Data Center services going south ↔ east (example: DHCP)
- R1 ↔ Data Center ↔ Peers going south ↔ east ↔ west ↔ north (example: IPv4 Internet traffic)
TERASTREAM HOMEGATEWAY PLATFORM

Existing CPE procurement models are far too slow to keep up with the necessary rate of development.

The current state of IPv6 support on commercial platforms leaves ... room for improvement!

We believe that we are not the only carrier with this problem.

**So ...**

DT are currently developing a carrier grade’ CPE software platform.

Based on OpenWRT, extended with IPv6 and SP functionality.

Plan to register this with Linux Foundation.

A fully open source initiative.

We are looking for other interested carriers to build a development community around the platform.
IPv6
IPv6 NATIVE NETWORKING
IPv6 is fundamental, not an Afterthought

A new network - designed from the ground up specifically for IPv6
IPv6 for ALL internal interfaces (incl. management, control plane etc) – with just a bit of IS-IS for IGP
Treat IPv6 as a new protocol – It can be much more than just ‘IPv4, but bigger’
Don’t let native IPv4 into the network
  No Dual-Stack
  Treat IPv4 as a long-tail overlay service – (more on this later)
  Once you let ANY IPv4 into the design, you’ll never get rid of it!
SERVICE DIFFERENTIATION BASED ON ADDRESSES USING IPv6 ADDRESS SPACE AS LABELS

Provider (56 bits)  User Subnet (8)  User – Host (64 bits)

Registry/IANA Assigned  Servicebits  Network Structure bits

P Public  0=SP-intern, 1=extern
I Infrastructure  0=end user, 1=infrastructure packet
E Endpoint/Service  0=endpoint, 1=service
SSS Service Type  0=res, 1=internet, 4=video, 5=L2, 6=voice, 7=mgmt
M  0=fixed, 1=mobile endpoint

Examples:

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIESSS</td>
<td>PIESSS</td>
</tr>
<tr>
<td>User  -&gt; IMS</td>
<td>000110</td>
</tr>
<tr>
<td>IMS  -&gt; User</td>
<td>011110</td>
</tr>
<tr>
<td>User  -&gt; User (best effort)</td>
<td>X00001</td>
</tr>
<tr>
<td>User  -&gt; Internet (best effort)</td>
<td>100001</td>
</tr>
<tr>
<td>Internet  -&gt; User (best effort)</td>
<td>XXXXXX</td>
</tr>
<tr>
<td>Lan-Lan service</td>
<td>010101</td>
</tr>
</tbody>
</table>
MULTIPLE PREFIXES ON THE CLIENT…

Brings benefits, but also a new (old!) problem

How do you ensure that the client selects the right source address for each different service?

Current source address mechanisms are based on variants of longest source/destination prefix matching policy

- This places constraints on your addressing architecture
- Often require additional policy to be provisioned to the client
- Doesn’t give users or applications information about the ‘properties’ or ‘suitability’ of a prefix for use

The Solution? - “Prefix Colouring”

- Adds additional metadata to DHCP prefix allocations
- Allows applications and users to select a source prefix based on this metadata
- Source address selection is decoupled from the destination address prefix matching
- Can also help with source address selection in multi-homing deployments
- Described in ‘draft-lepape-6man-prefix-metadata’
In this example, two different services are being run on the same network. The service provider wishes that traffic is sourced from different prefixes by the home network clients for Video on demand service as against general Internet access.

The homenet has several prefixes delegated – (potentially one each for voice, video and Internet)

This example shows prefix classes for different services:
- Prefix Class: Internet (e.g. 0)
- Prefix Class: Video (e.g. 10)
IPv4 AS A SERVICE – LIGHTWEIGHT 4o6 SOFTWIRES

Lightweight 4o6 Softwire Tunnel Concentrator (lwAFTR)
**If Not IPv6, use the network as a PTP Ethernet**

Scenario: *New customer connects to DTIP Infrastructure*
1) Customer connects his device, generates IPv6 packet and forwards them to R1
2) R1 receives the IP packet and recognizes a VLAN as not “in service”
3) IP packet is encapsulated at R1, with the IPv6 informing the VLAN / L2 device from where the request came from, and forwards it to Data Center
4) Data center responds with DHCP or Web server. Customer can only reach the walled garden.

**Scenario: Customer registers**
1) Web server at Data Center generates a request to OSS to configure a new customer via NetConf / Yang at router R1, Line ID.
2) The OSS via NetConf configures the R1 as “in service” for a customer located at a specific interface (IPv6 address).
3) From now on, the customer is outside the walled garden and can reach other Internet addresses.
There are still mainstream products that do not have complete IPv6 implementations:
- Transport interfaces tend to have more complete implementations
- Management and control plane functionality may not be so good

The level of IPv6 testing in shipping products is not on a par with IPv4 – we’ve found some pretty hairy bugs!

Vendor’s need constant ‘encouragement’ to resolve these problems

If you are planning any kind of similar rollout, get your requirements fixed and test well in advance!
NETWORK FUNCTION VIRTUALISATION

“THE INFRASTRUCTURE CLOUD”
INFRASTRUCTURE CLOUD
NETWORK FUNCTION VIRTUALIZATION

OTT Apps
Self Provisioning
IPv4 Softwire
Business VPN Services

IMS
Video
Network Services (DNS, DHCP)

Mobile Core & Services

Content

40% of traffic

R2
REAL-TIME OSS
Define “Services” in a data-model language (Yang)

NCS CLI: EVPL Service over R1 and R2

```
ncs# services service evpl2 type evpl bandwidth 100MB qos-template gold port-1 r1-sto0-ge0/0/1.65 port-2 r1-gbg1-ge0/0/5.65
ncs(config-service-evpl2)# show
ncs(config-service-evpl2)# show conf
services service evpl2
type evpl qos-template gold
type evpl bandwidth 100MB
type evpl port-1 r1-sto0-ge0/0/1.65
type evpl port-2 r1-gbg1-ge0/0/5.65
!
cs(config-service-evpl2)# commit
```

NCS REST

```
$ curl -v -X POST -T evpl.xml -u admin:admin \ http://localhost:8080/api/running/services/service
```

NCS NETCONF

```
$ netconf-console --edit-config=evpl.xml
```
TERASTREAM
CROATIA TRIAL
100 Gb/s network using IP and Optical integration

Full integration of Network and Cloud technologies for service production

Native IPv6 network delivering consumer service

Built in a record time - decision in September, launch on Dec 10th

500 customers with up to Gigabit access speeds

Agile execution – small cross-functional teams (DT, HT, Cisco, Combis)

Continued development and iterative improvement
  Technology refinements
  New vendors being integrated
  More customers connected
Croatia expansions in 2013:

- Expand TeraStream pilot in Croatia
  - Increase the network
  - Integrate in HT environment
  - Improve data center
### TeraStream

- Radical simplification of the IP network architecture
- Removing the legacy from the core (IPv4, MPLS), improving services
- Optical transmission is integrated into the IP routers using 100G coherent technologies
- Combining network and cloud for scalable service production
- Control using a SDN paradigm – Realtime OSS

### Benefits

- Improve user experience, real Internet services to more users
- Use just enough complexity to do the job and no more
- Get the revenue and cost balance right
Questions?

Now you can bring out your tar and feathers and start throwing things at me..

THANKS!