

NTT's IP Out-Of-Band (OOB) management network

Job Snijders

NTT GIN / AS 2914

IP Development Engineer

job@ntt.net

Out of band networks

[...] can be used to ensure management connectivity [...] *independent* of the status of other in-band network components.

All you need to ensure is that the out-of-band network does not depend on the in-band network (aka *production*)



When do you use an Out-Of-Band network?

- **Routine: during planned router maintenance**
 - When the router software is to be upgraded
 - When physical components are added/removed from the chassis
 - During turn-up, turn-down of the POP, or expansions
- **Incidents: when there is a problem with the in-band network**
 - Did the device crash?
 - Was there a hardware problem: did a linecard fail?
 - Is there some kind of physical connection issue, all fiber uplinks cut?

NTT operates multiple BGP Autonomous Systems

The primary Autonomous System number is 2914, NTT's Global IP Network, which you of course all know and love!

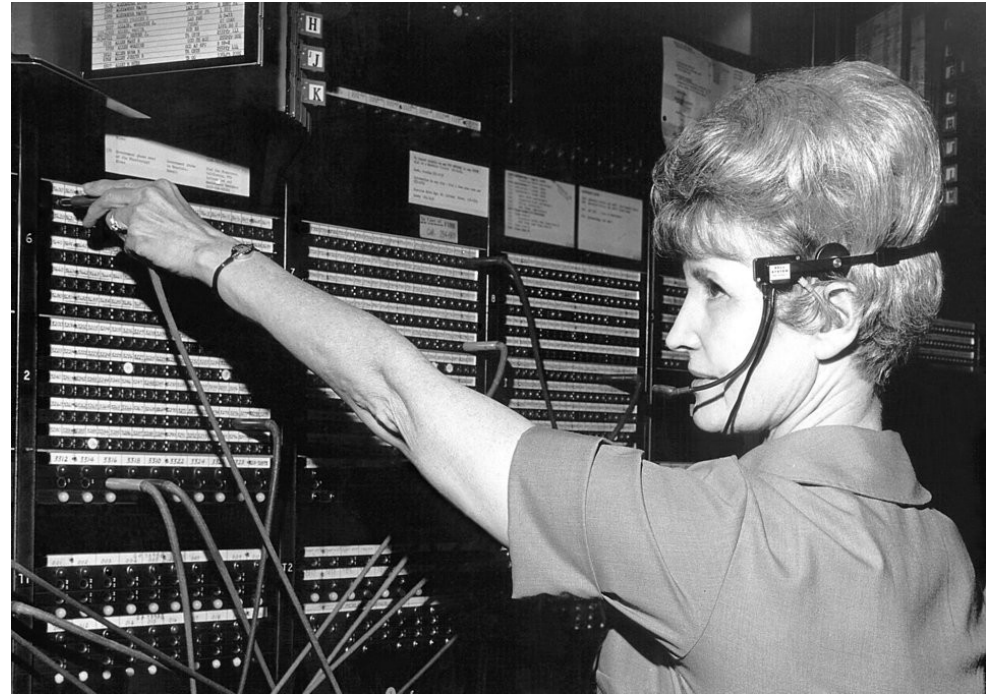
However a second *separate* AS, on running *separate* routers, which operate under a *separate* routing policy, and which use commercial BGP IP transit services to connect to the Internet also exists to support out-of-band management.

NTT's out-of-band network (2016, colorized)

Phone line connected to modem,
modem connected to RS-232 port

The operator will dial out via POTS,
and hope a connection would
establish....

... hope is not a strategy



A modern approach to OOB: using IP via the Internet

NTT's OOB requirements

- All management modules of all devices *permanently* connected via RS-232 via SSH via IP via IPSEC via IP all the way back to the mothership
- All RS-232 interfaces concurrently available (with the POTS network it was 1 at a time)
- All serial traffic continuously logged (archived) for debugging purposes
- Multi-user: NTT staff to concurrently work on the same RS-232 CLI terminal lines
- Simplicity in OOB device configuration: all POPs follow a *cookie stamp* design
- Flexibility for OOB WAN uplink, use whatever connection we have
 - Cellular IP (4G simcard), IP via DIA on Electrical or Optical interfaces

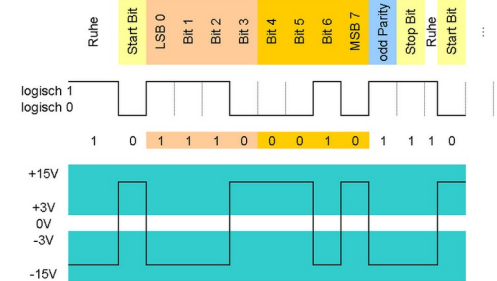
Rough Bill Of Materials (BOM)

- Cisco *ISR4331-SEC/K9*
- 1 or 2 *NIM-24A* modules (24 RS-232 interfaces each)
- Breakout cables *CAB-ASYNC-8* (three per *NIM-24A*)
- 1 x WAN module (Cellular, Optical or Electrical)
- Software license, support, memory, flash, etc
- Juniper *EX2300* for extending the OOB network inside a POP



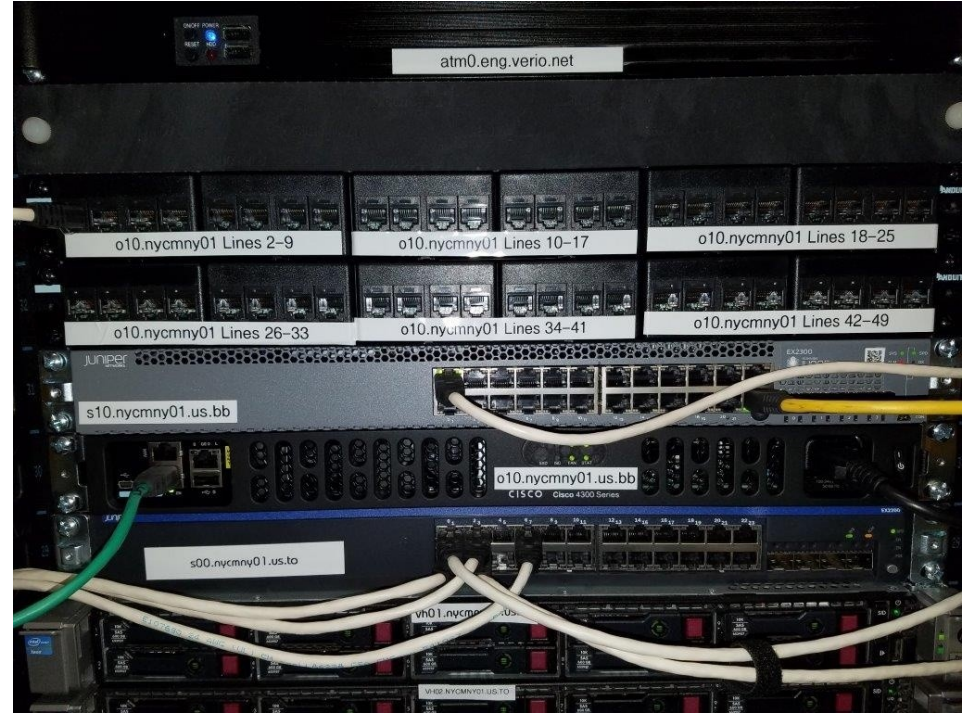
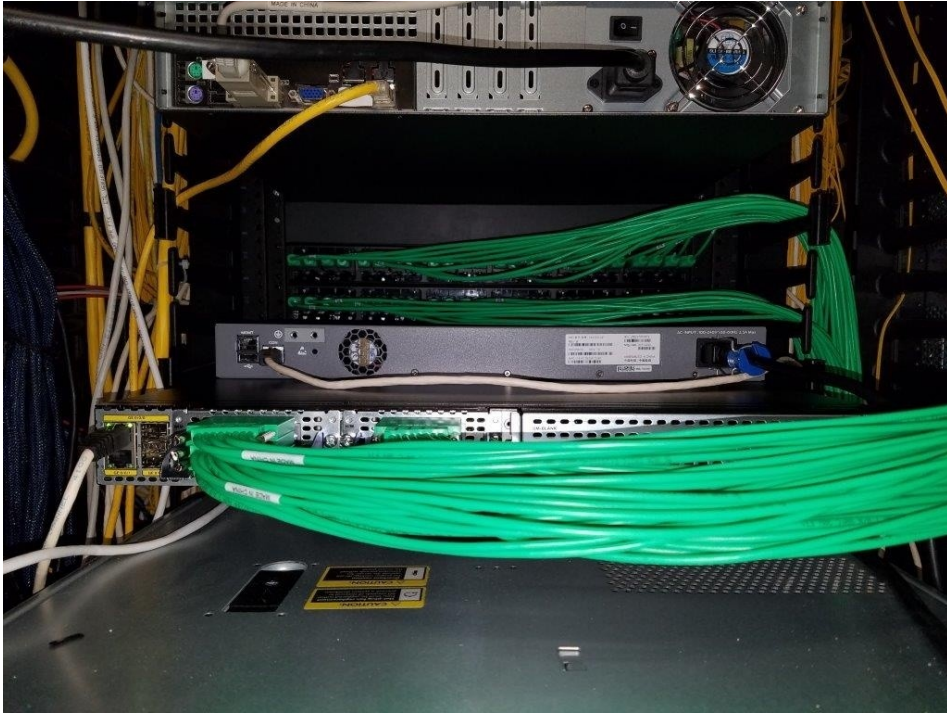
Synchronisation
Daten low & high
Check

9600 8O1 = 9600 Baud; 8 Datenbits; odd Parity; 1 Stopbit
ASCII "G" = \$47 = 0100 0111

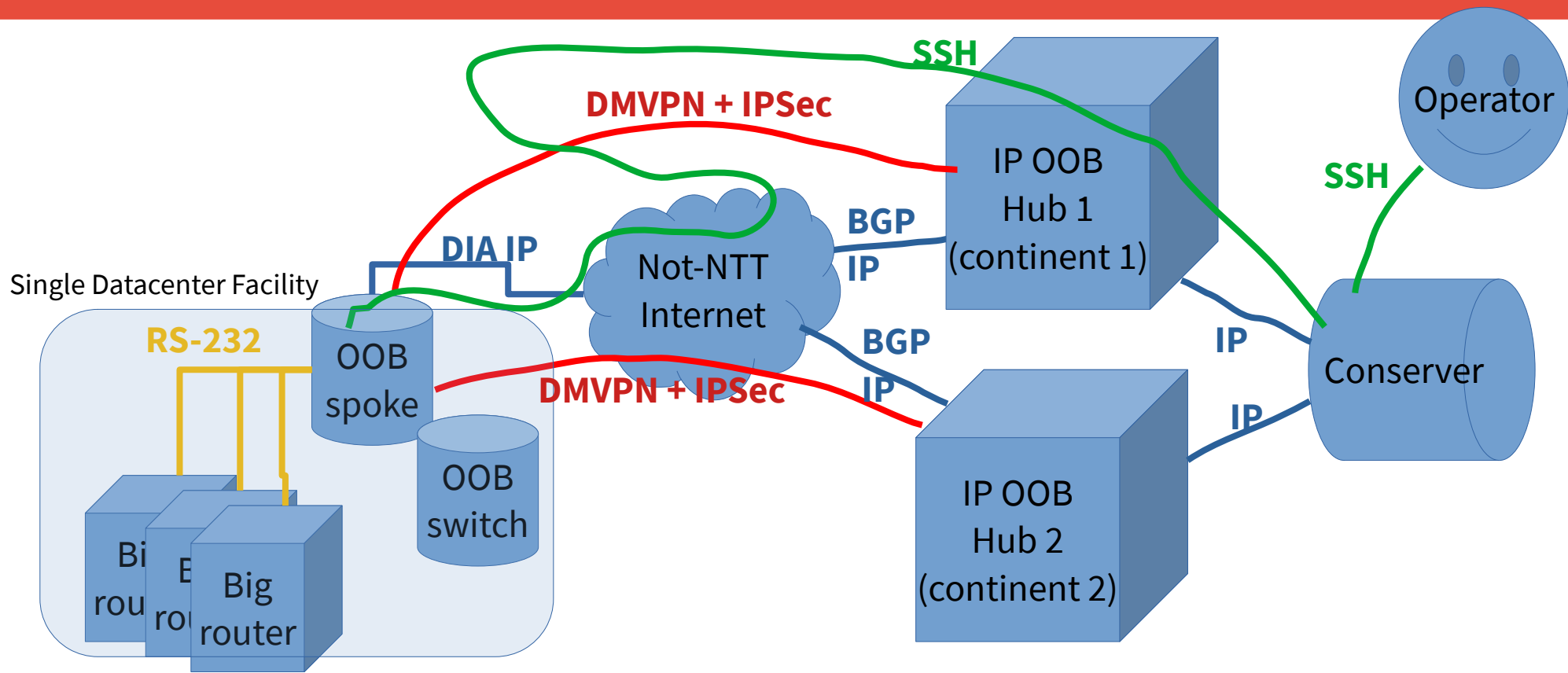


Yes, this really is a modern RS-232 heavy solution!!! :-)

What it can look like (2020, colorized)



Out of Band Network topology template



Global routing considerations

- The OOB DMVPN Hubs operate under a *different* ASN than GIN's 2914 number, as the Hubs' BGP routing policy is different than AS 2914's.
- Each DMVPN Hub announces its own globally unique prefixes to the Default-Free Zone
- The DMVPN Hubs are connected to specific selected non-NTT providers
- Each DMVPN Spoke is connected to specific selected non-NTT provider
- Configs on Hub + Spoke are static, but (re-)routing is dynamic (DMVPN, NHRP, BGP, ISIS)

“non-NTT” means that to the best of our understanding, the provider has no dependency on NTT to be able to transport packets between the Hubs and Spokes of the NTT IP OOB network.

Tying it all together: Conserver

Conserver is magic



https://www.conserver.com

Home Download Mailing Lists Documentation ▾

- Log output for forensics...
- Multiple simultaneous users...
- Flexible access control...
- Secure communications...

```
$ console universe
Enter bryan's password:
[Enter '^Ec?' for help]
[replay]
crash dump...
...
...
[disconnect]
$
```

What is conserver?

Why is conserver so important?

- NTT manages thousands of RS-232 connections, need some type of organization
- Hard for humans to memorize port numbers / interface numbers: use names
- *Conserver* creates abstraction layer between terminal and 'how to connect'
- You can connect by name, and *conserver* will figure out how to connect you to the RS-232 terminal
- Through *conserver*, a *separate* daily monitoring program performs various checks to automatically confirm whether the RS-232 connection is (still) correctly connected.

What does using conserver look like?

```
$ console r02.amstnl02.nl.bb
console: ambiguous console abbreviation, `r02.amstnl02.nl.bb'
      choices are r02.amstnl02.nl.bb_rsp1, r02.amstnl02.nl.bb_rsp0
$ console r02.amstnl02.nl.bb_rsp1
[Enter `^Ec?' for help]
NTT Communications                               Unauthorized Access is Prohibited
r02.amstnl02.nl.bb.gin.ntt.net                   For Service Call +1-877-688-6625
                                                    20200719-008

User Access Verification
Username: [disconnect]
$
```

Periodic maintenance of the Out-of-Band network

The DMVPN/IPSec sessions, BGP sessions, and SSH connections are easy to monitor, but RS-232 also requires monitoring.

NTT wrote a RS-232 regression monitoring tool “*badmeester*” which (at daily intervals) connects to every RS-232 endpoint and tests whether the other side is still properly connected.

Pro-actively end-to-end testing each and every RS-232 connection managed through *conserver*, means we know exactly *before* we undertake Big Router maintenance operations, which serial lines are working and which are not working as expected! Very useful.

Conclusion – NTT has a really cool SD-WAN OOB solution!



Give as much thought, investment, and attention to your out-of-band management network as you do to your production network!