# NTT Global IP Network Configuration Tools Overview
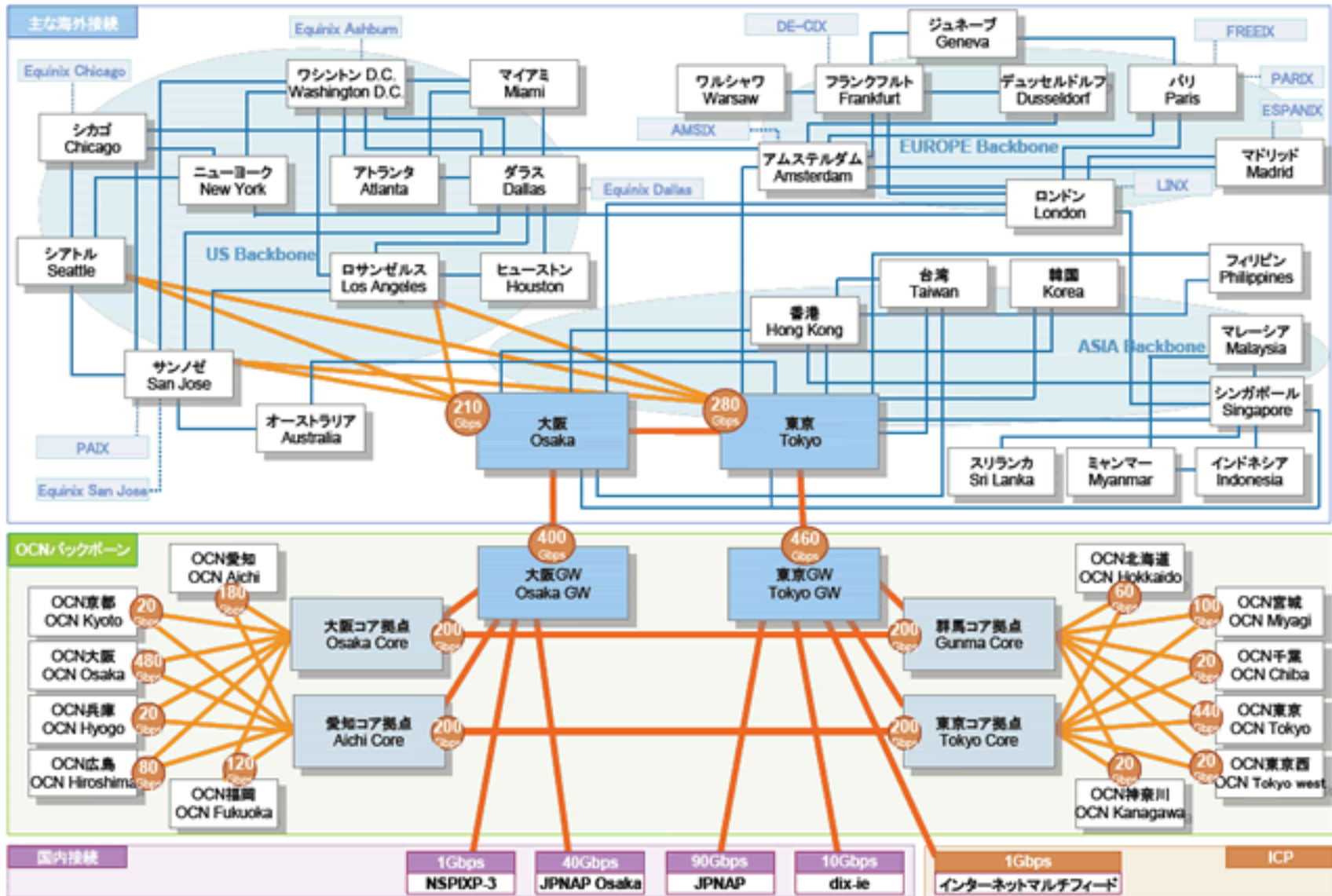
Presented by:
ショーン　モリス shawn@ntt.net
有賀 征爾 say@ntt.net
NTTアメリカ
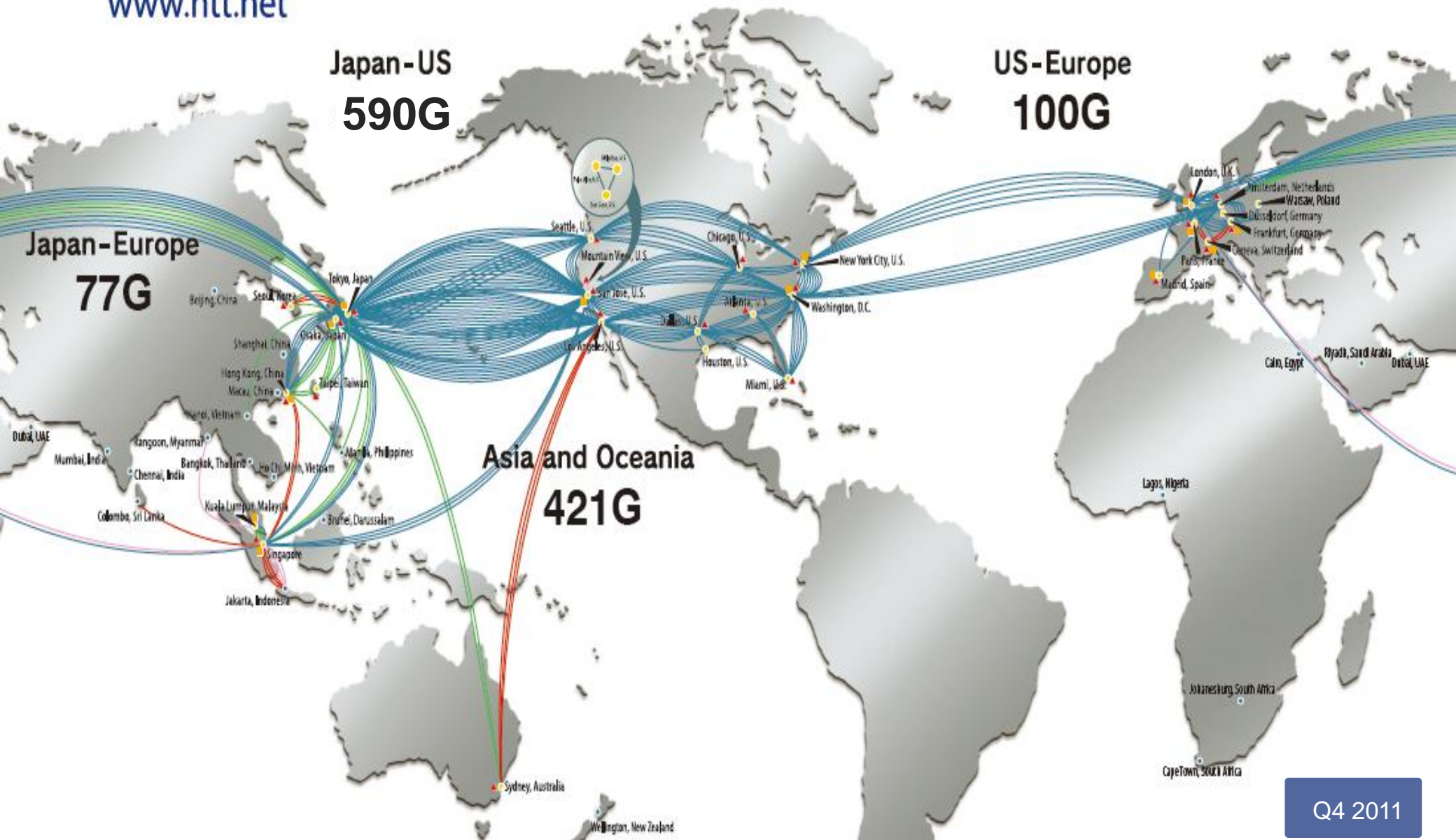
# What is the NTT Global IP Network?
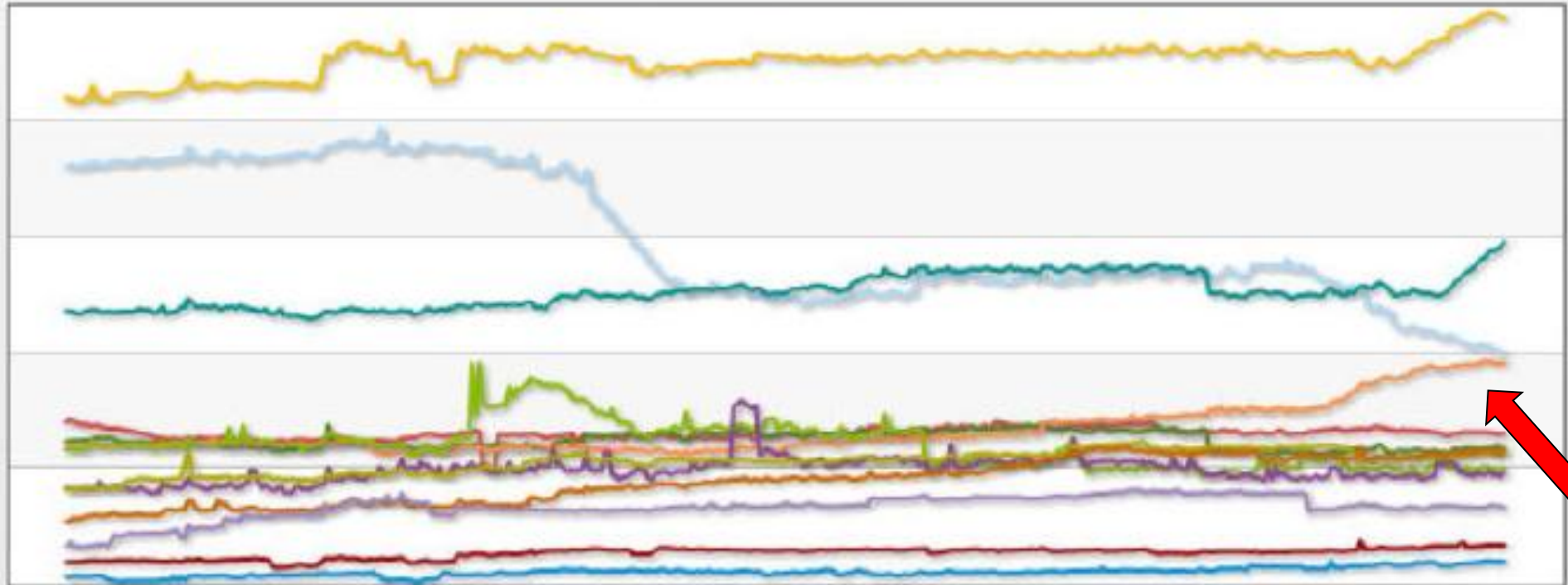
# NTT Global IP Network (AS2914)

Baker's Dozen for 2010

Legend: Level 3, Sprint, Global Crossing, Savvis, TeliaSonera, NTT, Verizon, AT&T, Tata, Tinet, China Telecom, Cogent, Qwest

## NTT GIN Philosophy

- IP Transit pricing experiences a consistent downward pressure.
  - Underlying costs must be managed in a similar fashion
- Operating Expenses are kept low through a large degree of automation

## Configuration Tools Overview

- SQL database driven
- SQL-derived data is transformed through platform specific templates
- Automated deployment of changes
- All configuration is driven from server
  - No persistent manual configuration on devices

- Original M4 templates created by Randy Bush at rain.net.

- 1997, Verio added additional functionality based on irrtoolset.

- 1998, SQL database and Perl code were added

- Over time data was migrated from flat files to SQL

- Eventually perl became a bottleneck and was replaced with C code (including custom m4 parser).

- PostgreSQL

- GNU Make

- M4 macros

- bgptool homegrown binary
  - Includes custom M4 processor

- Custom scripts built on RANCID for pushing configurations to routers

- Standard ACLs
- Standard device parameters
  - AAA configuration
  - SNMP
  - Logging
- Interface parameters
- Routing policy

- Can include router OS version dependent options

- SSH access
- Ability to retrieve files via FTP
- Commit/roll back/roll forward capapility
- Ability to directly manipulate the startup configuration
- Text-based configuration storage

| Management | Interfaces | MPLS | Routing |
|---|---|---|---|

| SNMP Communities | ASN Information | Standard ACL | OOB Ports | IP Addressing | Virtual Interfaces | LAG | Ethernet | MAC Accounting | SONET | LDP | RSVP | RSVP-TE LSP | PWE | ISIS | OSPF | BGP |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

SQL Database      M4 Macros      Config File      Router
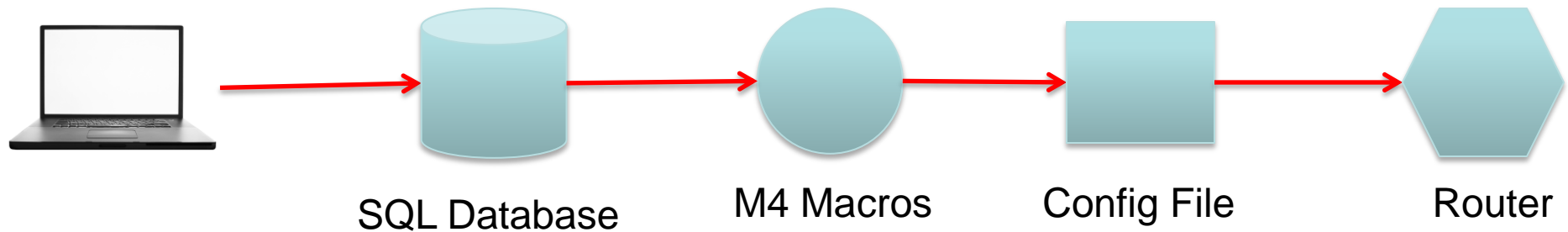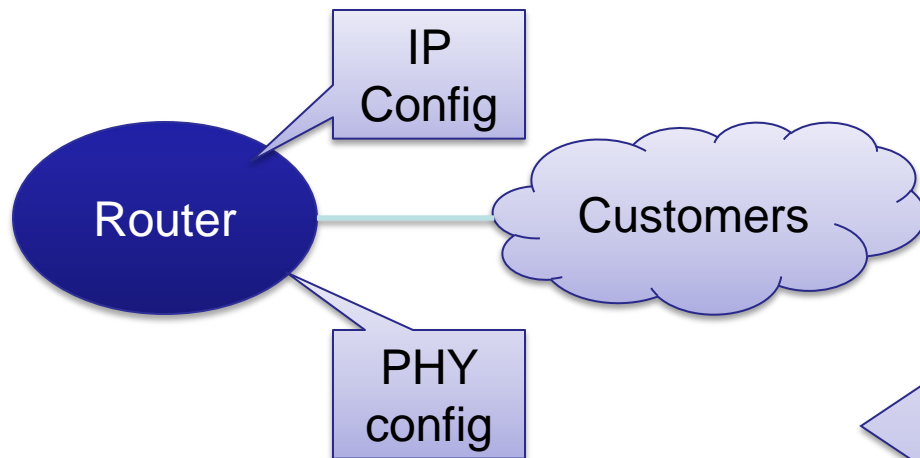
1. User enters config changes via Web UI or CLI
2. User initiates config build via UNIX make command on configuration server
3. User initiates config push via UNIX command
   1. Router is contacted by script via SSH
   2. Router requests configuration from server via FTP
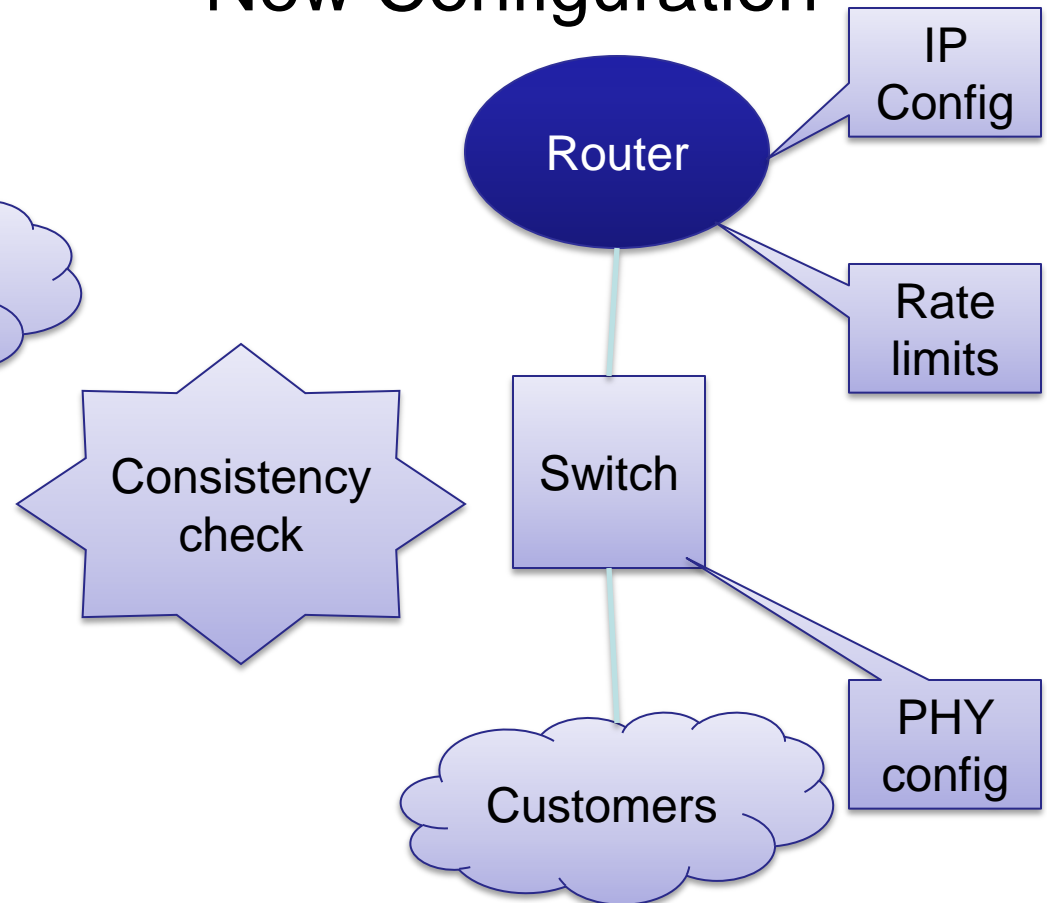   3. Configuration is committed

- Lower staffing requirements
- Lower error rates
  - Lower risk of catastrophic errors
- Rapid service delivery
  - Rapid service migration
- Automated service delivery
- Minimize peer review

- Configuration consistency
- Reporting

- **Automatic customer BGP ACL and max prefix updates**
- **Tool to load peer/customer configurations on IOS**
  - All relevant config (ie. interface, BGP neighbor, policy) is loaded via 'copy ftp: running-config
- **Mass update of RSVP-TE LSPs**
- **Bulk move of interfaces/sub-interfaces**
- **Seeding of other systems with data (stats system, monitoring system, etc.)**
- **Complex configurations across multiple devices**

- Current configuration
- New Configuration

IP Config

Router

Customers

PHY config

Consistency check

IP Config

Router

Rate limits

Switch

PHY config

Customers

- Integration with OSS system
- Linecard/module awareness

14

# Demonstration by
## 有賀さん

# Questions?