OpenFlow: Enabling Innovation in Campus Networks

Nick McKeown
Stanford University

Presenter: Munhwan Choi

Table of contents

- What is OpenFlow?
- The OpenFlow switch
- Using OpenFlow
- OpenFlow Switch Specification
- OpenFlow consortium
- Conclusion

What is OpenFlow?

- OpenFlow is an open standard that enables researchers to run experimental protocols in the campus networks.
- OpenFlow is added as a feature to commercial Ethernet switches, routers and wireless access points.
- OpenFlow provides a standardized hook to allow researchers to run experiments, without requiring vendors to expose the internal workings of their network devices.

Innovations in campus wiring closets

- Experiments we'd like to do
  - Mobility management
  - Network-wide energy management
  - New naming/addressing schemes
  - Network access control
- Problem with current network
  - Paths are fixed (by the network)
  - IP-only
  - Addresses dictated by DNS, DHCP, etc
  - No means to add new processing

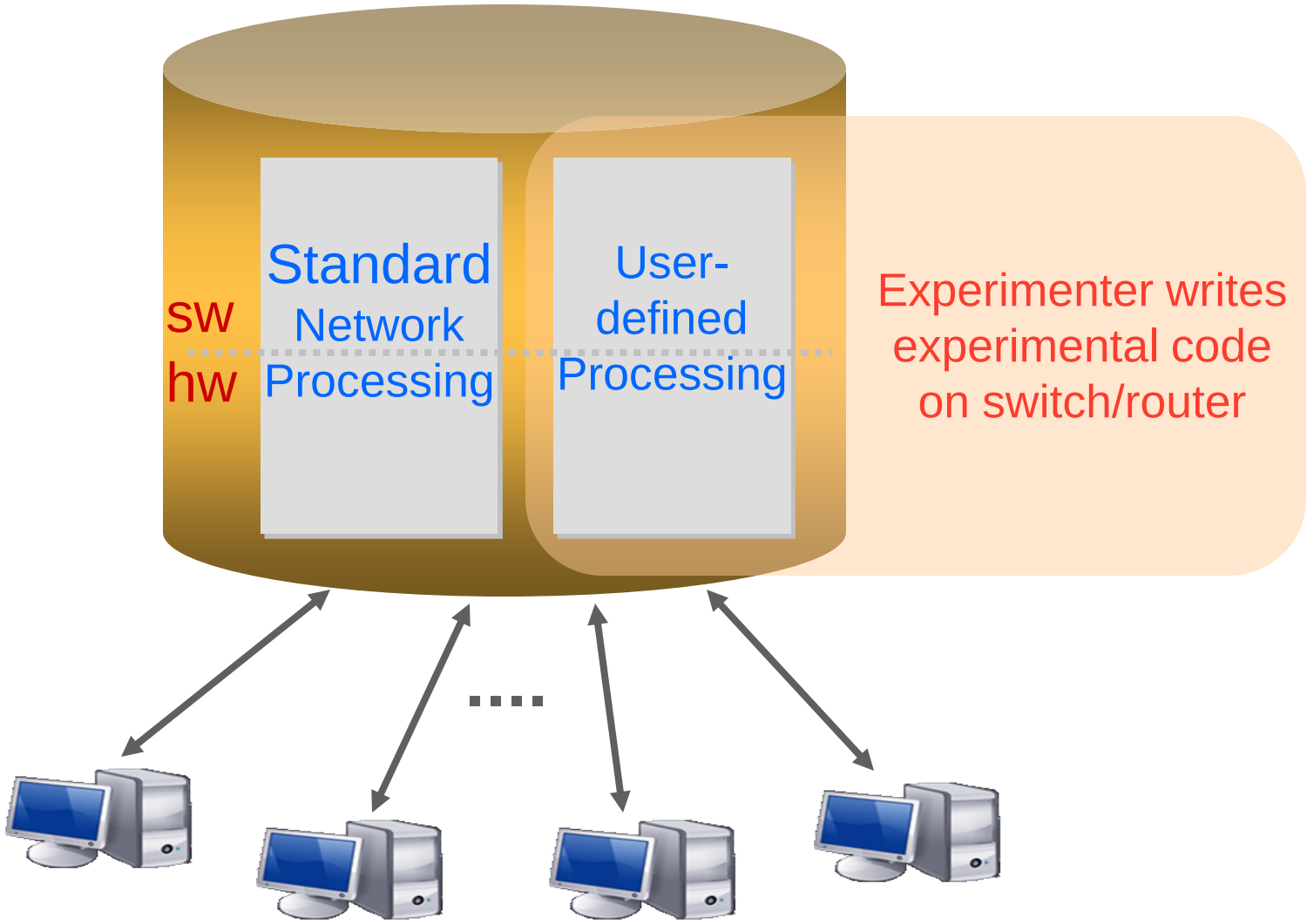
OpenFlow Switching

- A way to run experiments in the networks used everyday.
- Bring GENI to college campuses.

- A "pragmatic" compromise
  - *Allow researchers to run experiments in their network…*
  - *…without requiring vendors to expose internal workings.*

- Basics
  - *An Ethernet switch (e.g. 128-ports of 1 Gigabyte Ethernet)*
  - *An open protocol to remotely add/remove flow entries*

Experimenter's Dream (Vendor's Nightmare)



No obvious way

- Commercial vendor won't open software and hardware development environment
  - Complexity of support
  - Market protection and barrier to entry
- Hard to build my own
  - Prototypes are flakey
  - Software only: Too slow
  - Hardware/software: Fanout too small (need >100 ports for wiring closet)

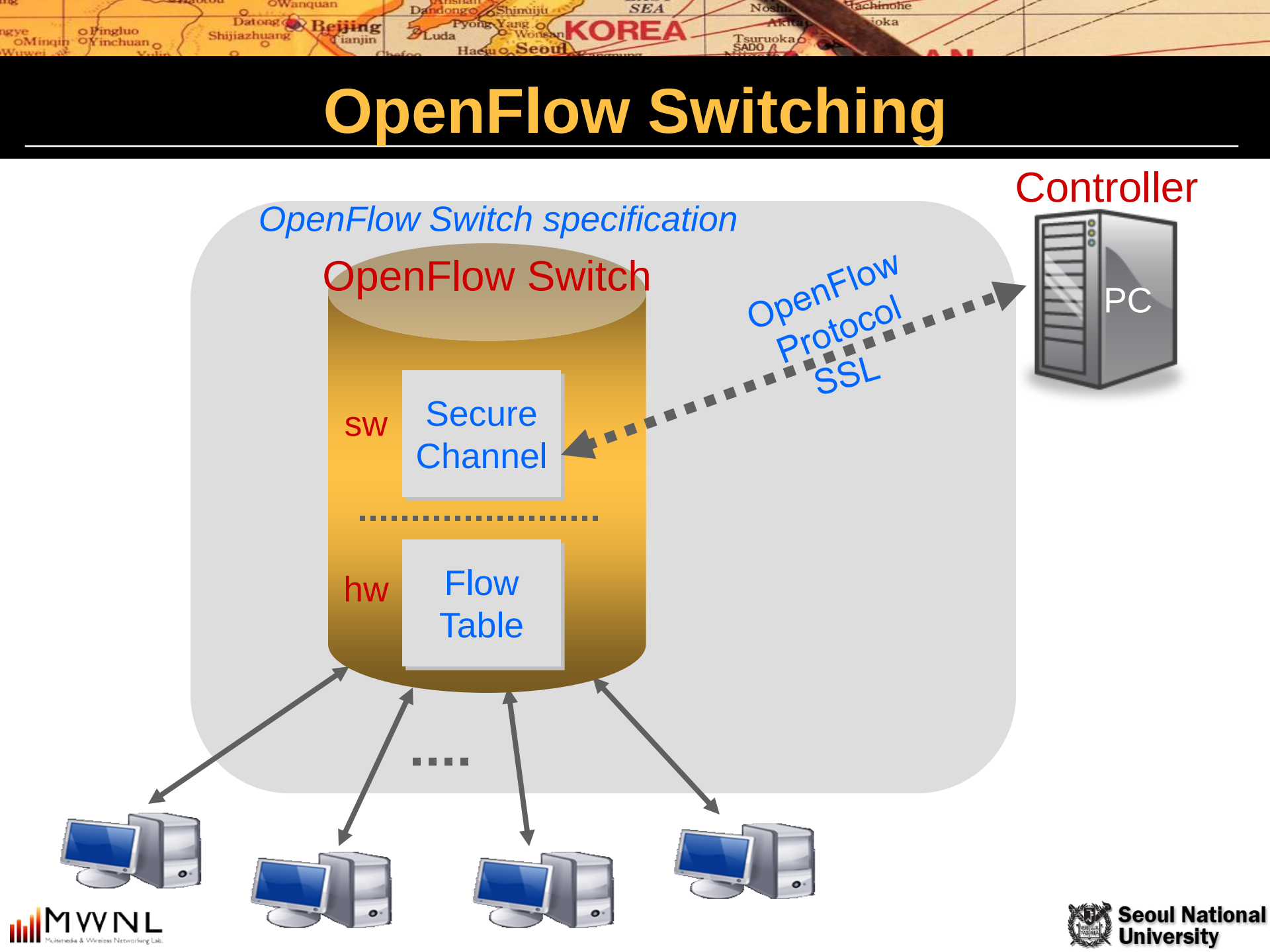
Furthermore, we want…

- Isolation: Regular production traffic untouched
- Virtualized and programmable: Different flows processed in different ways
- Equipment we can trust in our wiring closet
- Open development environment for all researchers (e.g. Linux, Verilog, etc).
- Flexible definitions of a flow
  - Individual application traffic
  - Aggregated flows
  - Alternatives to IP running side-by-side
  - …

THE NEED FOR PROGRAMMABLE NETWORKS

- Amenable to high-performance and low-cost implementations.
- Capable of supporting a broad range of research.
- Assured to isolate experimental traffic from production traffic.
- Consistent with vendors' need for closed platforms.

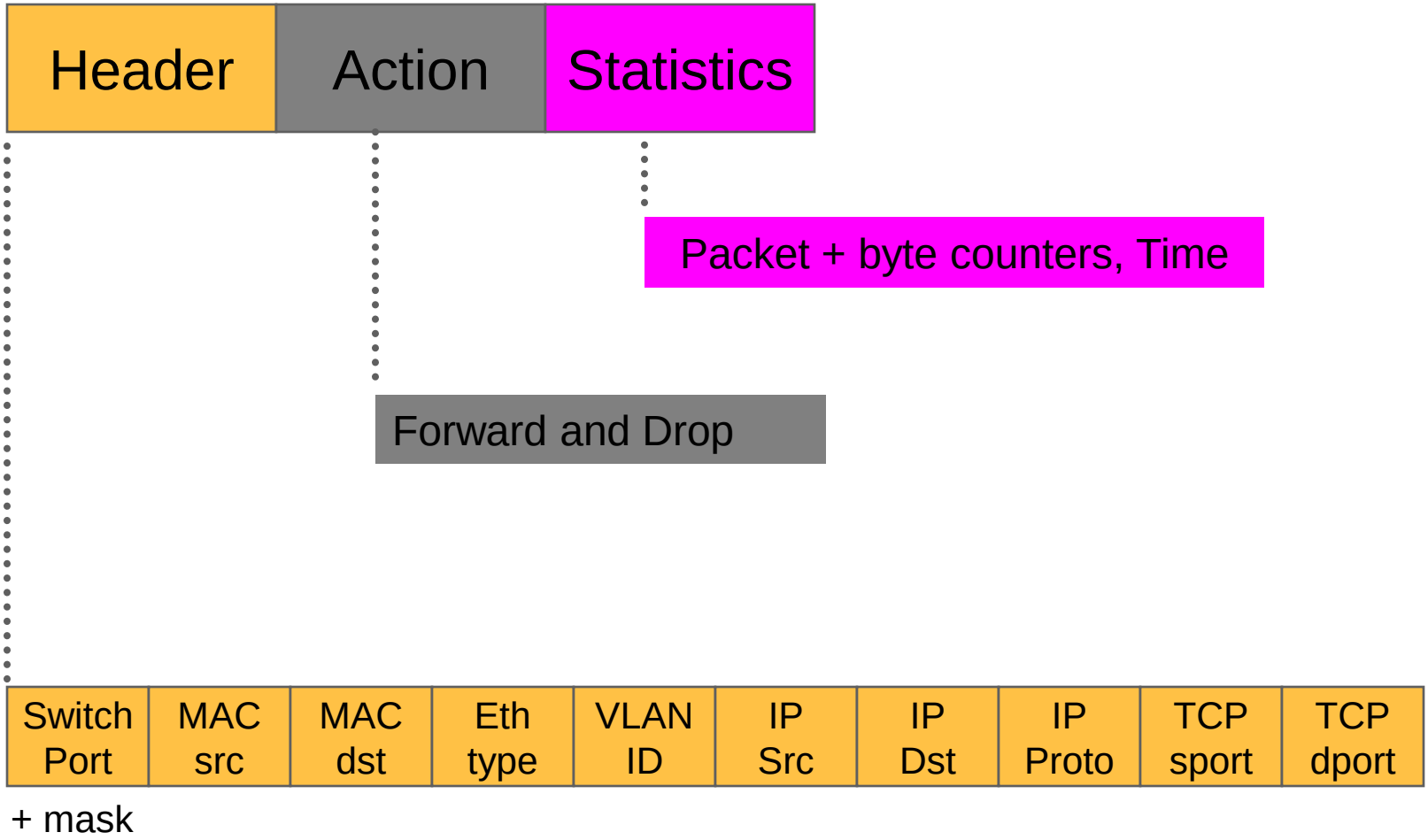
OpenFlow Switching



Two types OpenFlow Switches

- Type 0
  - The minimum requirements for any conforming OpenFlow Switch.
- Type 1
  - Superset of Type 0, and remain to be defined.

Flow Table Entry - “Type 0” OpenFlow Switch

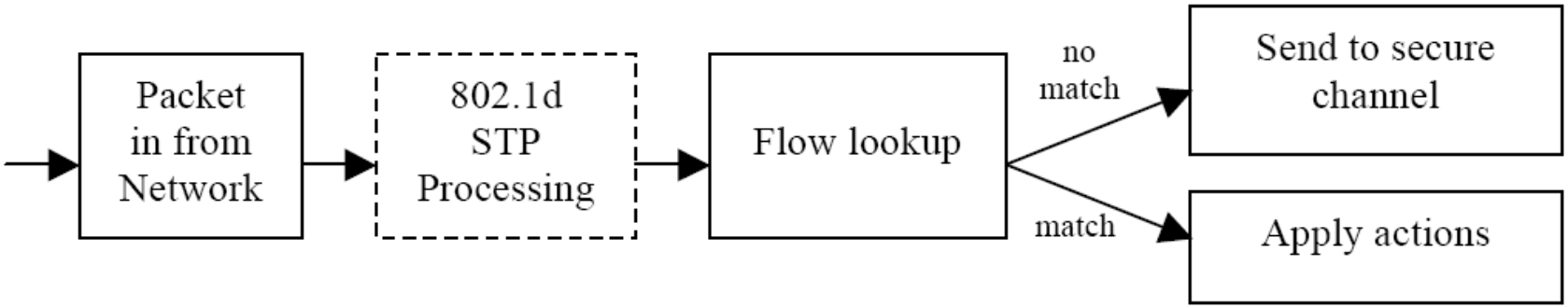


Four Actions

- Dedicated OpenFlow switches.
  1. Forward to external port(s)
  2. Encapsulate and forward to controller over Secure Channel
  3. Drop packet
- OpenFlow-enabled switches
  4. Forward to the normal forwarding path (normal processing pipeline) of this switch
     - e.g. for normal Layer 2 and Layer 3 processing.

Dataflow

- On receipt of a packet, an OpenFlow Switch performs the functions.
  - Rules specifying an ingress port are matched against the physical port that received the packet.
  - The Ethernet headers are used for all packets.
  - If the packet is a VLAN (Ethernet type 0x8100), the VLAN ID is used in the lookup.
  - For IP packets (Ethernet type equal to 0x0800), the lookup fields also include those in the IP header.
  - For IP packets that are TCP or UDP (IP protocol is equal to 6 or 17), the lookup includes the transport ports.

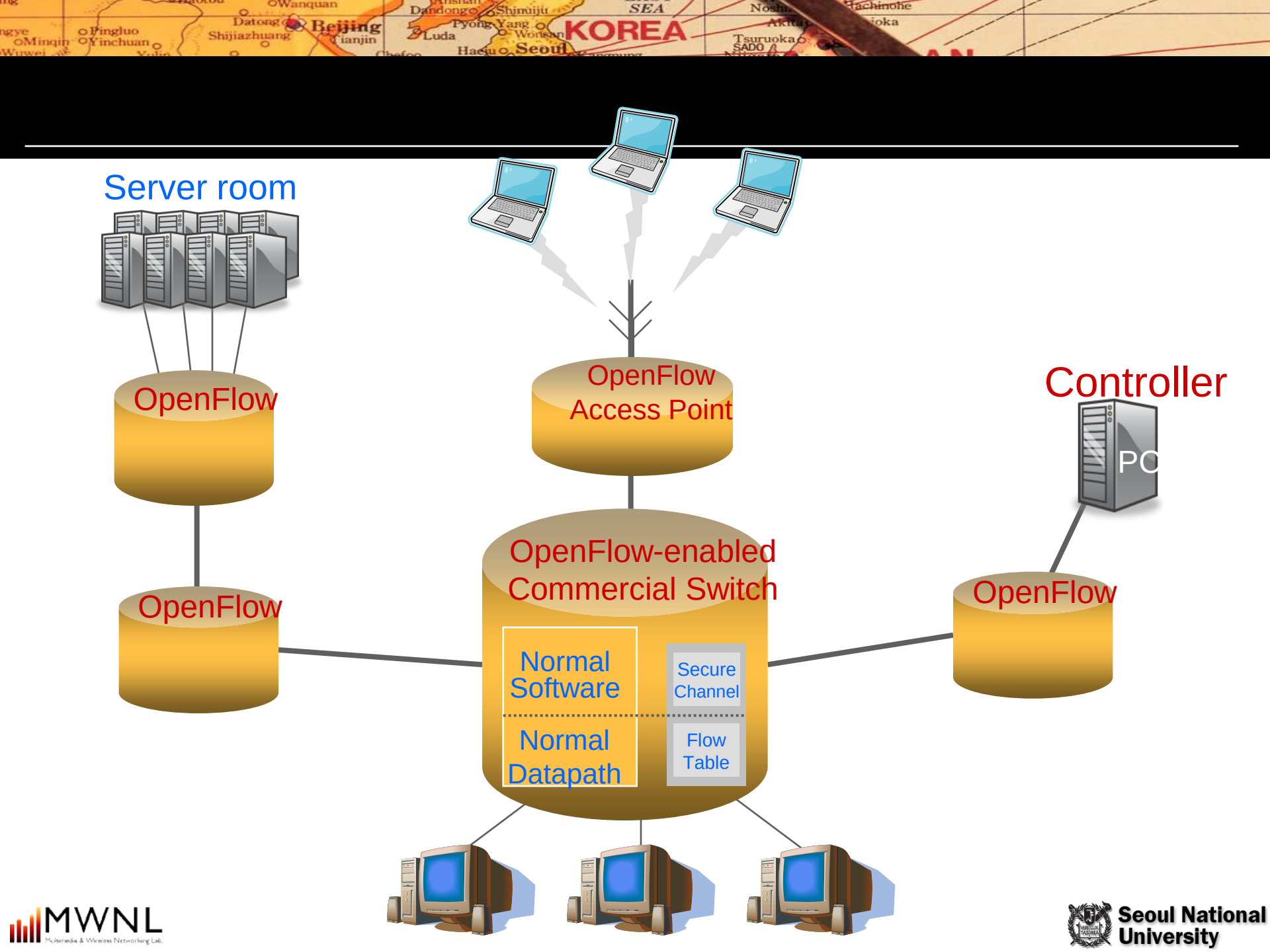


OpenFlow "Type 1"

- Definition in progress
- Additional actions
  - Rewrite headers
  - Map to queue/class
  - Encrypt
- More flexible header
  - Allow arbitrary matching of first few bytes
- Support multiple controllers
  - Load-balancing and reliability

Secure Channel

- SSL Connection, site-specific key
- Controller discovery protocol
- Encapsulate packets for controller
- Send link/port state to controller


Server room
OpenFlow
OpenFlow
OpenFlow
Access Point
OpenFlow-enabled
Commercial Switch
Normal
Software
Normal
Datapath
Secure
Channel
Flow
Table
Controller
PC
OpenFlow

OpenFlow Usage Models

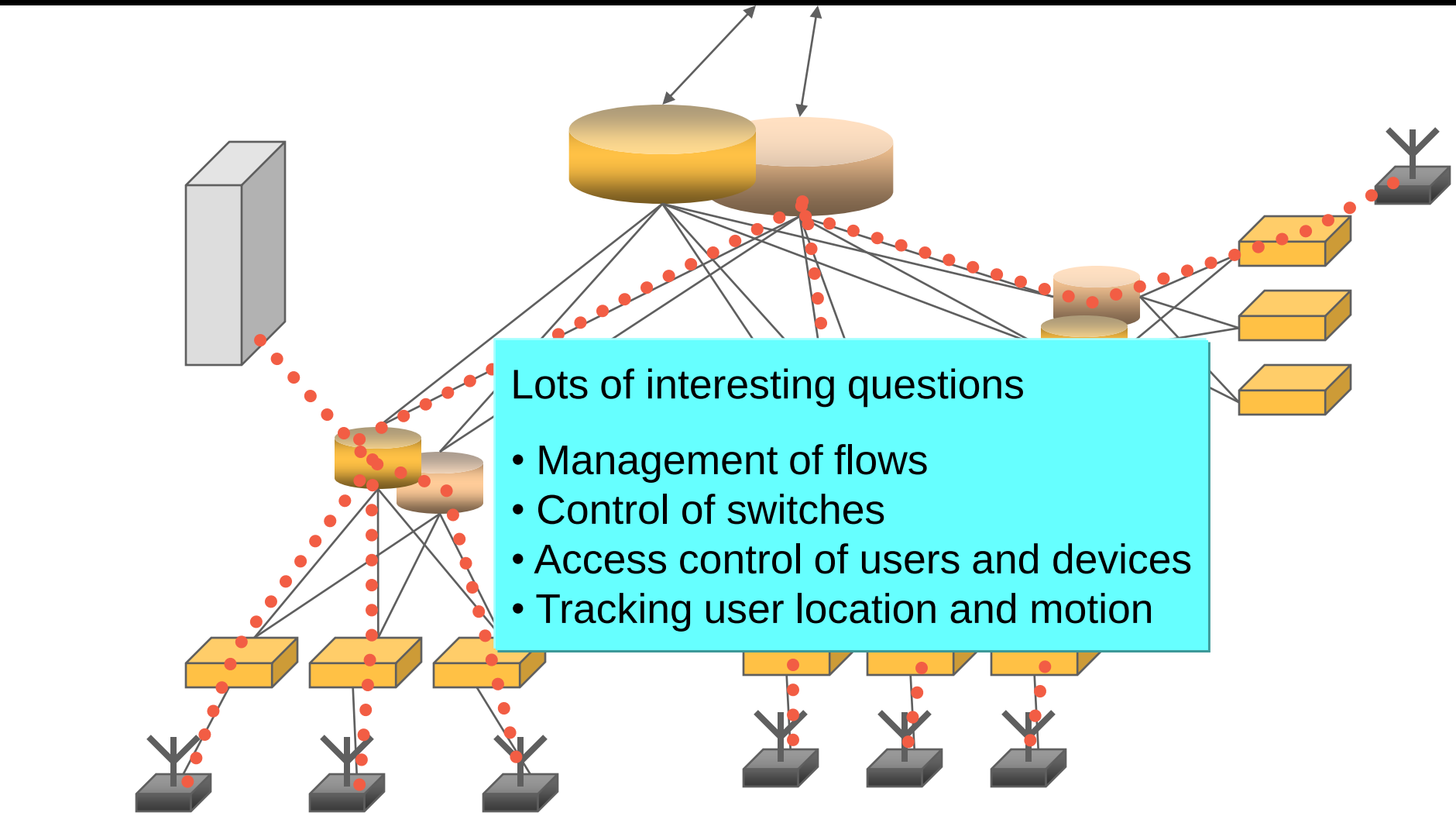
- Experiments at the flow level
  - User-defined routing protocols
  - Admission control
  - Network access control
  - Network management
  - Energy management
  - VOIP mobility and handoff
  - …

  → Experiment-specific controllers
  → Static or dynamic flow-entries

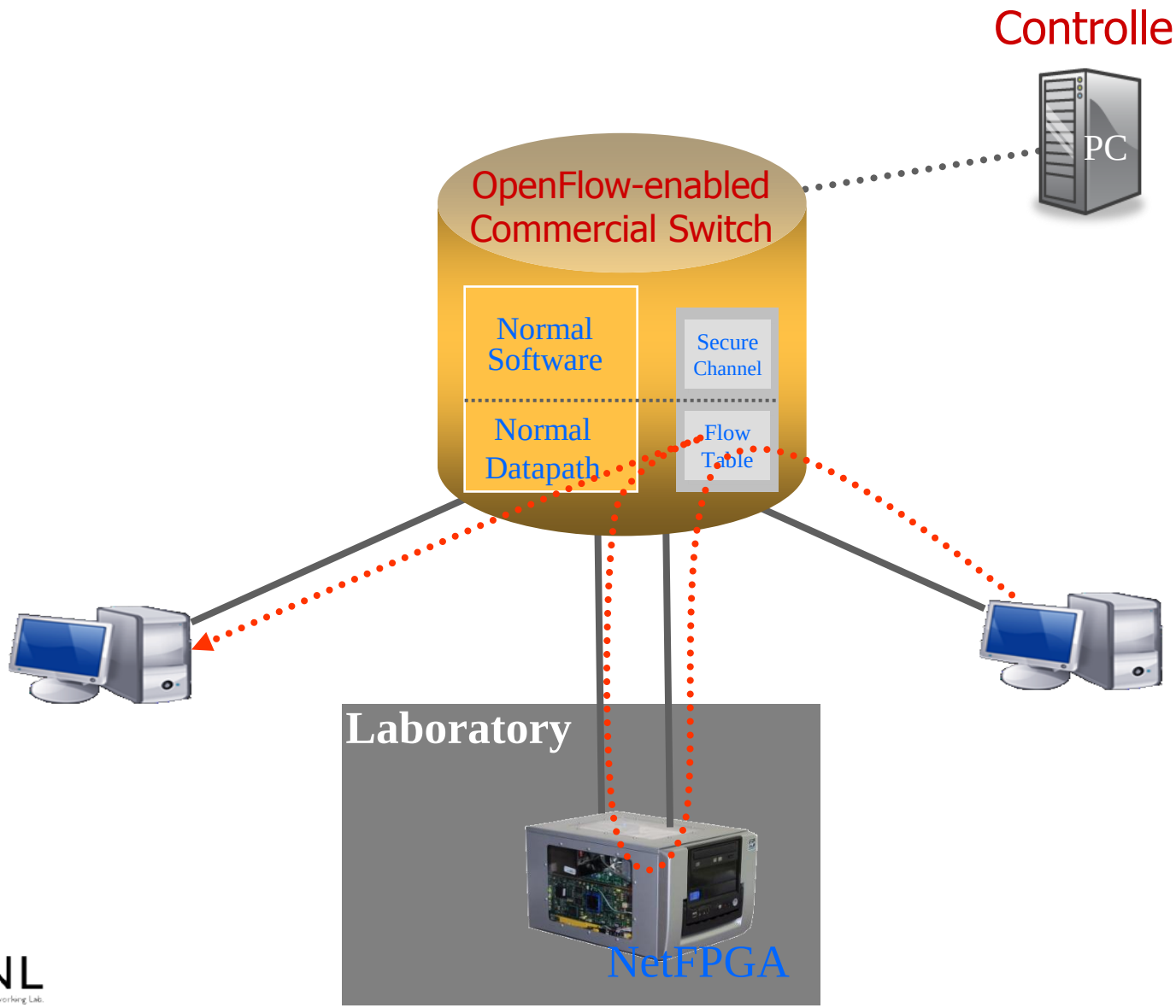
- Experiments at the packet level
  - Slow: Controller handles packet processing
  - Fast: Redirect flows through programmable hardware
  - Modified routers, firewalls, NAT, congestion control…

- Alternatives to IP

Example Experiment at the flow level - *Mobility*



Experiments at the packet level



OpenFlow Usage Models

1. Experiments at the flow level
2. Experiments at the packet level

3. Alternatives to IP
   - Flow-table is Layer-2 based
   - e.g. new naming and addressing schemes
   - …

OpenFlow Switch Specification

- OpenFlow Switch Specification,Version 0.8.1 (Draft)
  - The standards document that describes the protocol that is used between an OpenFlow Switch and the OpenFlow Controller.
  - Cover the components and the basic functions of the switch, and the OpenFlow protocol to manage an OpenFlow switch from a remote controller.

OpenFlow Consortium

- http://OpenFlowSwitch.org
- **Goal**: Evangelize OpenFlow to vendors
- Free membership for all researchers
- Whitepaper, OpenFlow Switch specification, Reference Designs
- Licensing: Free for research and commercial use

OpenFlow: Status

- Commercial Ethernet switches and routers
  - Working with six vendors to add to existing products
  - Expect OpenFlow "Type 0" to be available in 2008-09
- Reference switches
  - Software: Linux and OpenWRT (for access points)
  - Hardware: NetFPGA (line-rate 1GE; available soon)
  - Working on low-cost 48-port 1GE switch based on Broadcom reference design
- Reference controller
  - Simple test controller
  - NOX controller (Martin Casado; available soon)

Deployment at Stanford

- Stanford Computer Science Department

  Gates Building

  ~1,000 network users

  23 wiring closets

- Stanford Center for Integrated Systems (EE)

  Paul Allen Building

  ~200 network users

  6 wiring closets

Working with HP Labs and Cisco on deployment

Conclusion

- Enabling innovation on campus
- Standard way to control flow-tables in commercial switches and routers
- Being deployed at Stanford

Reference

- OpenFlow: enabling innovation in campus networks
  - Nick McKeown, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, Jonathan Turner, SIGCOMM Computer Communication Review, March 2008
- OpenFlow Switch Specification,Version 0.8.1 (Draft)
- OpenFlow Power Point Presentation of Nick McKeown


Mandalgovi
Sair Usu
Dalandzadgad
DESERT
MONGOL (INNER MONGOLIA)
Bayan Obo
Wuyuan
Baotou
HEILONGJIANG (MANCHURIA)
Abagnar Qi
Jarud Qi
Linxi
Tongyu
Kailu
Harbin
Mudanjiang
Jilin
Shuangliao
Tonghao
Hunchun
Yanji
Tieling
Shenyang
Jinzhou
Wanquan
Anshan
Dandong
Kanggye
Shimuiju
EAST SEA
Najin
Ch'ongjin
Datong
Beijing
Tianjin
Shijiazhuang
Pyong Yang
Luda
Wonsan
KOREA
Seoul
Haeju
Kangmung
Pingluo
Yinchuan
Minqin
Wuwei
Yulin
Taiyuan
Zhongwei
Xingtai
Chefoo
Weihai
Dezhou
Inchon
Taejon
Taegu
Pusan
Kwangju
Qindao
Linfen
Jinan
Jining
Ya'an
Lanzhou
Longxi
Qingyang
Baoji
Huang (Yellow)
Kaifeng
Luoyang
Xuzhou
Lianyungang
Huai'an
Nanyang
Nancheng
YELLOW
Cheju
Svetlaya
Dalnegorsk
Turi Rogo
Ussuriysk
Vladivostok
Wakkanai
Esashi
ITURUP
KUNASHIR
SHIKOTAN
Abashiri
Otaru
Sapporo
HOKKAIDO
C. Erimo
Kushiro
Hakodate
Ominato
Hachinohe
Noshiro
Akita
Tsuruoka
SADO
Niigata
HONSHU
Kanazawa
Mt. Fuji
Nagoya
Tokyo
Taira
Yokohama
Kobe
Osaka
Wakayama
SHIKOKU
Fukuoka
Nagasaki
Yokoham

Thank You !