IIT KHARAGPUR

NPTEL ONLINE
CERTIFICATION COURSES

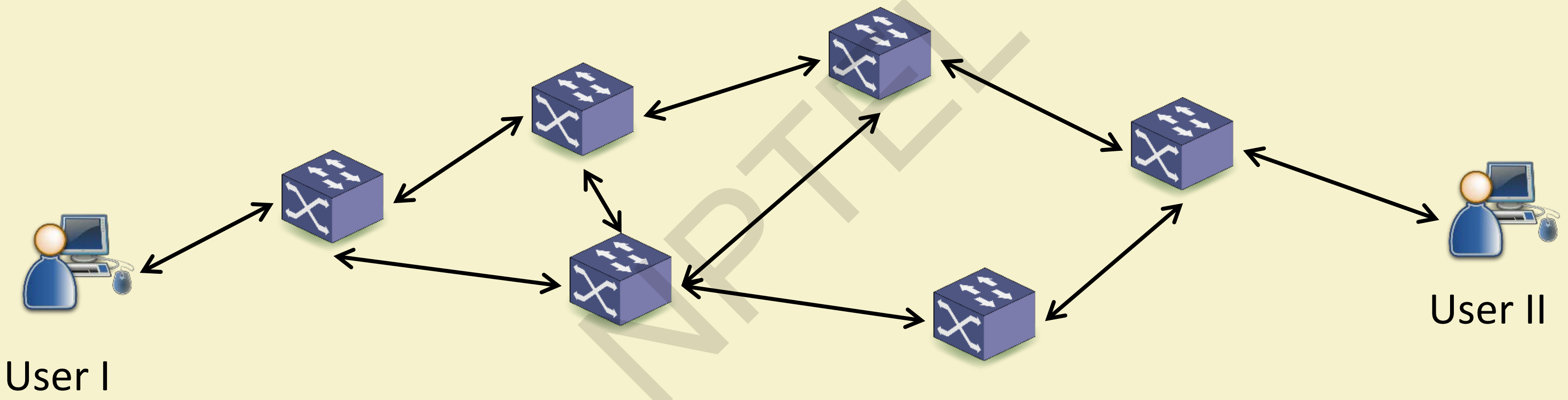
NPTEL

Software-Defined Networking – Part I

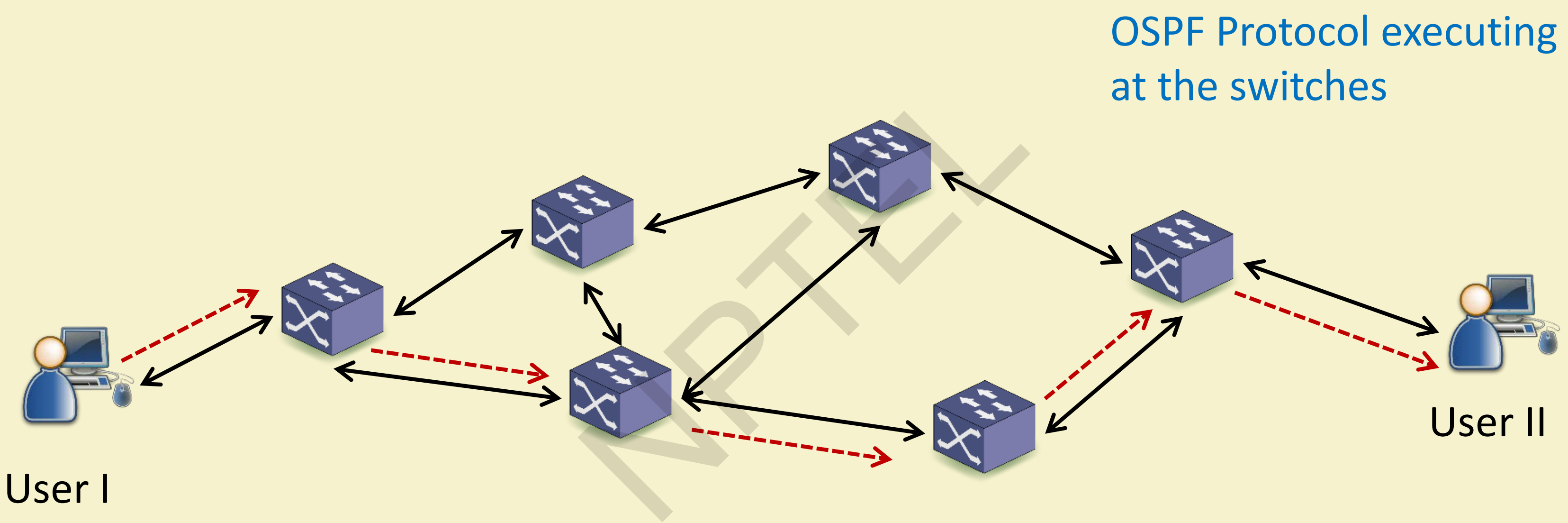
Restructuring the Current Network Infrastructure

Dr. Sudip Misra
Associate Professor
Department of Computer Science and Engineering
IIT KHARAGPUR
Email: smisra@sit.iitkgp.ernet.in
Website: http://cse.iitkgp.ac.in/~smisra/

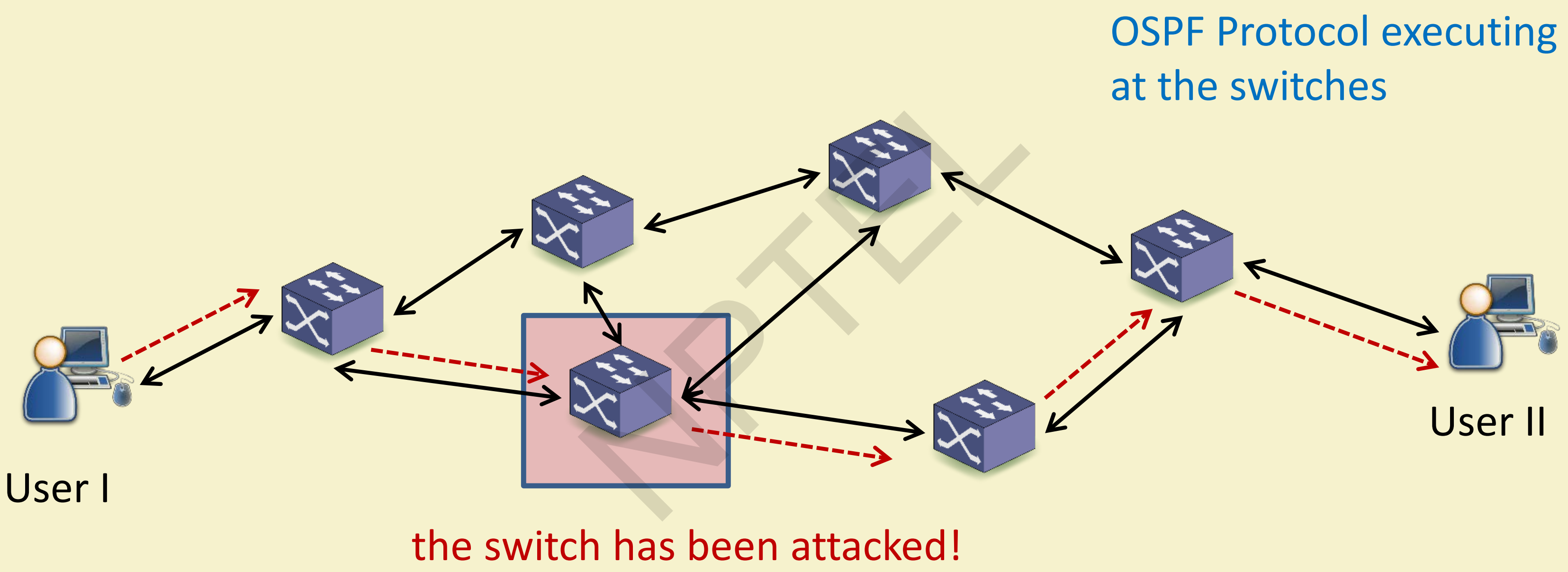
Overview of Current Network



Overview of Current Network

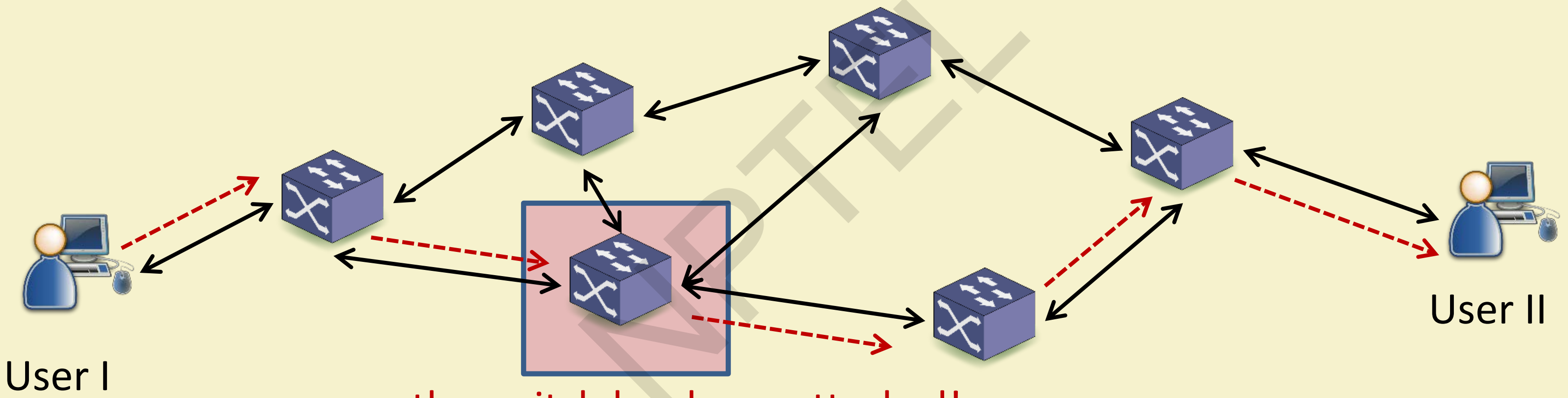


Overview of Current Network



Overview of Current Network

OSPF Protocol executes at the switches

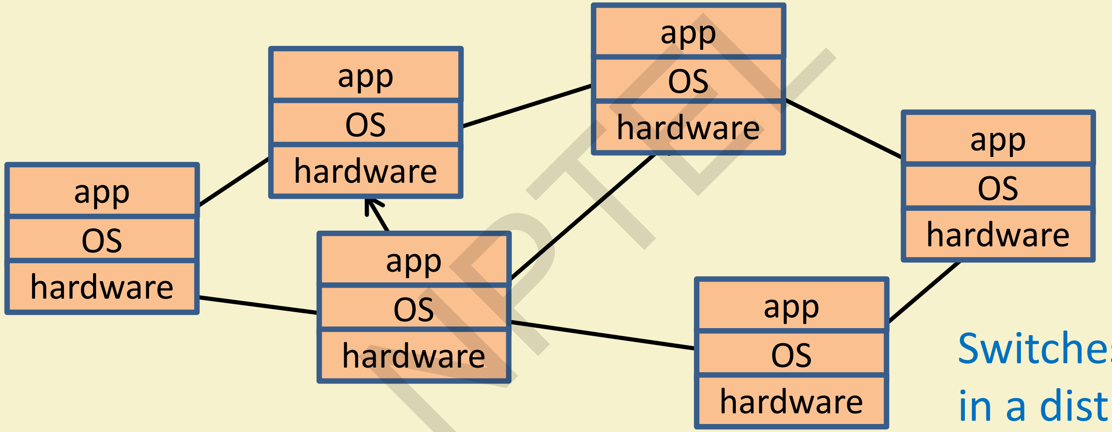


the switch has been attacked!

needs to route through an alternate path!

Present: No centralized control.

Limitations in Current Network

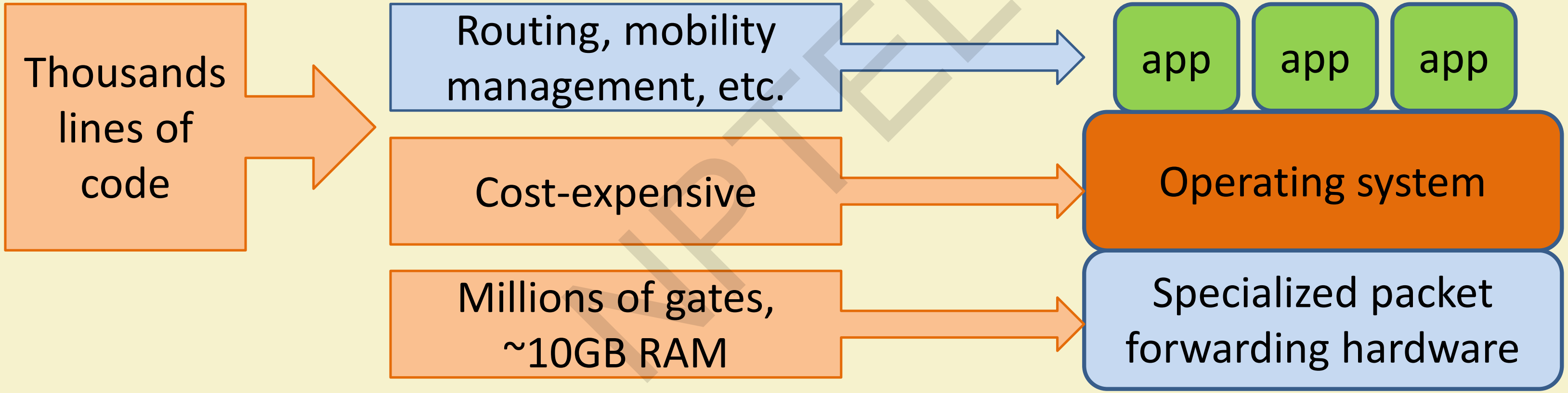


Switches forward traffic in a distributed manner. They do not have a global view of the network

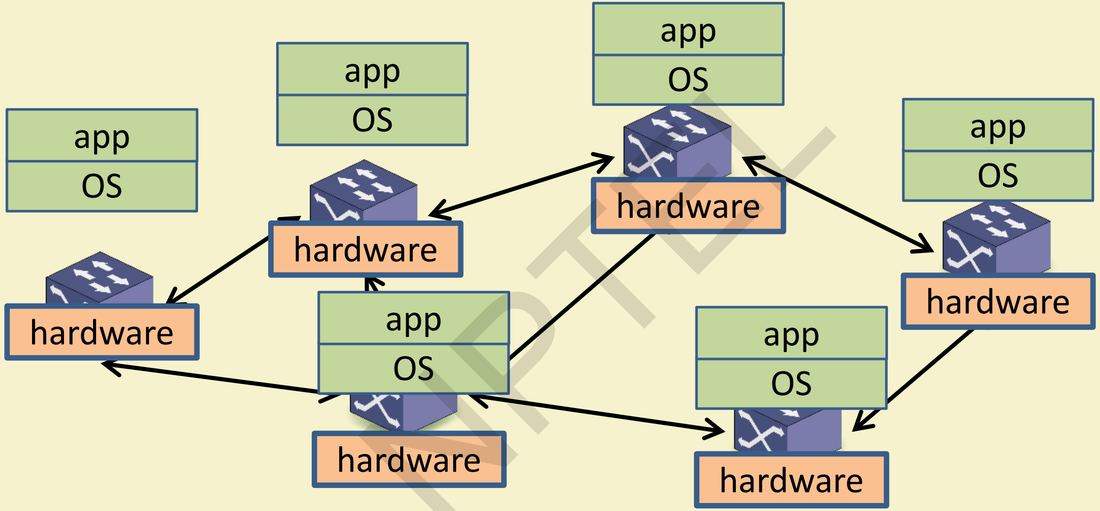
Limitations in Current Network

- ✓ Vendor-specific architecture of switches limits dynamic configuration according to application-specific requirements.
- ✓ Switches are required to configure according to the installed operating system (OS).
- ✓ Centralized control is not feasible in traditional network.

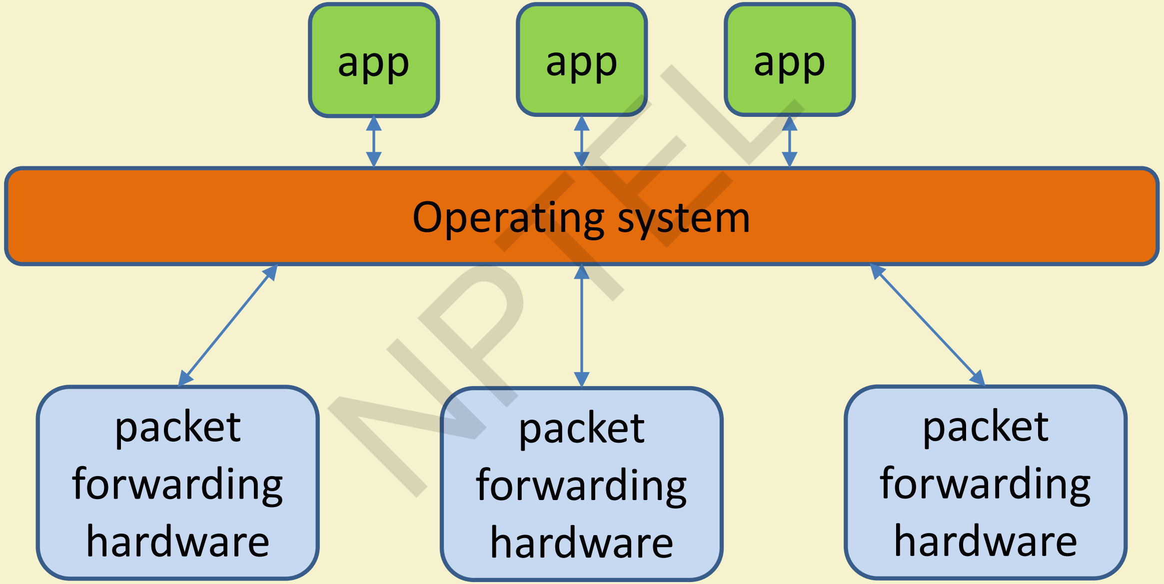
Limitations in Current Network



Current Network to SDN



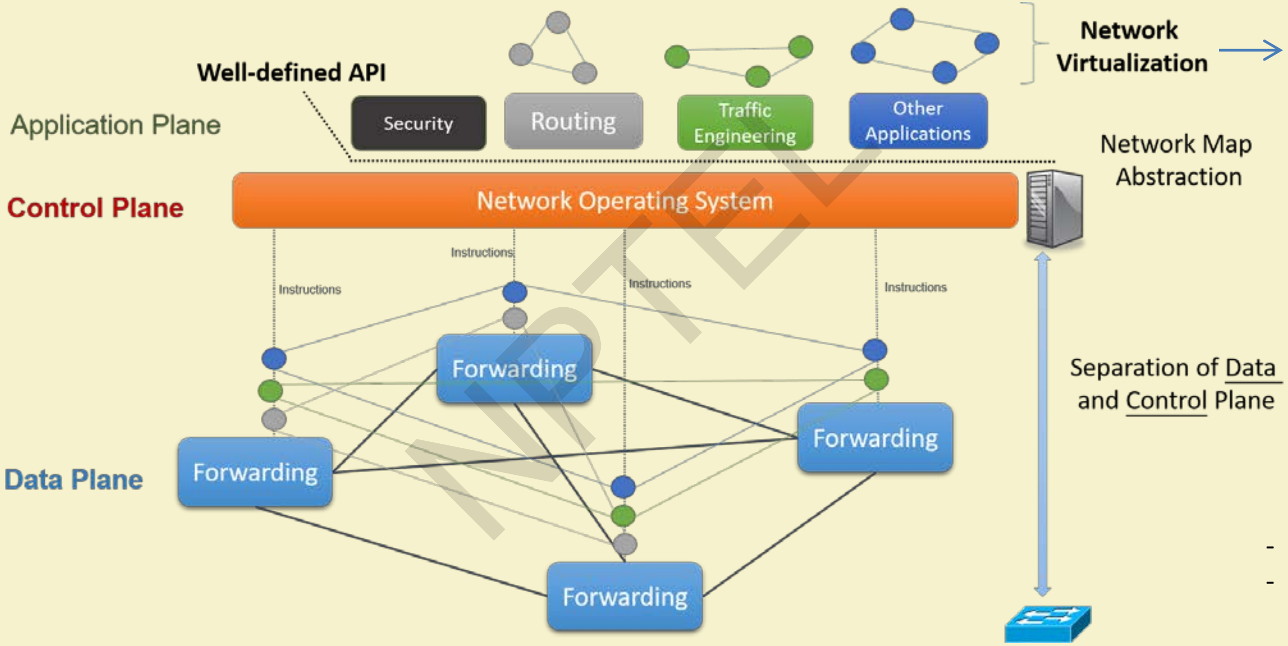
Current Network to SDN



Origin of SDN

- ✓ 2006: At Stanford university, a team proposes a clean-slate security architecture (SANE) to control security policies in a centralized manner instead of doing it at edges.
- ✓ 2008: The idea of *software-defined network* is originated from OpenFlow project (*ACM SIGCOMM 2008*).
- ✓ 2009: Stanford publishes OpenFlow V1.0.0 specs.
- ✓ June 2009: Nicira network is founded.
- ✓ March 2011: Open Networking Foundation is formed.
- ✓ Oct 2011: First Open Networking Summit. Many Industries (Juniper, Cisco announced to incorporate.

SDN Architecture



Network Function Virtualization

Notes
- Northbound API
- Southbound API (via OpenFlow)

Basic Concepts of SDN

- ✓ Separate control logic from hardware switches
- ✓ Define the control logic in a centralized manner
- ✓ Control the entire network including individual switches
- ✓ Communication between the application, control, and data planes are done through APIs

Components/Attributes of SDN

- ✓ Hardware switches
- ✓ Controller
- ✓ Applications
- ✓ Flow-Rules
- ✓ Application programming interfaces (APIs)

Current Status of SDN

- ✓ Companies such as Google have started to implement SDN at their datacenter networks.
- ✓ It is required to change the current network with SDN in a phased manner.
- ✓ Operational cost and delay caused due to link failure can be significantly minimized.

Challenges

- ✓ Rule placement
- ✓ Controller placement

Rule Placement I

- ✓ Switches forward traffic based on a rule – 'Flow-Rule' – defined by the centralized controller.
  - Traditionally, Routing Table in every switch (L3 switch/router). SDN maintains Flow Table at every switch.
  - Flow-Rule: Every entry in the Flow Table.
- ✓ Each rule has a specific format, which is also defined by a protocol (e.g., OpenFlow).

Rule Placement II

Match SDN Applications First and Use Normal For Unmatched Packets (Hybrid Default Forwarding)

Priority	Ingress Port	MAC Source Address	MAC Destination	Protocol	Vlan ID	IP Source Address	IP Destination	Source Port	Destination Port	Instructions
10000	*	*	*	TCP	*	*	10.1.1.20/32	*	80	Forward to Port 1
5000	*	*	*	*	*	*	10.1.1.0/24	*	*	Forward to Port 2
300	*	*	*	*	2600	*	*	*	*	Send to Controller
0	*	*	*	*	*	*	*	*	*	OF Normal

Example of a flow-rule based on OpenFlow protocol

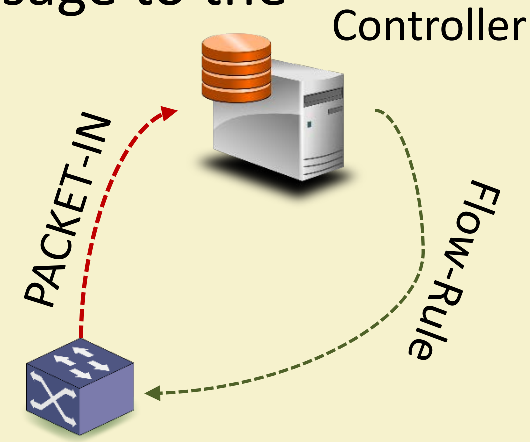
Source: http://networkstatic.net/wp-content/uploads/2013/06/OFP_normal_rules.png

Rule Placement Challenges I

- ✓ Size of ternary content-addressable memory (TCAM) is limited at the switches.
  - Limited number of rules can be inserted.
- ✓ Fast processing is done using TCAM at the switches.
- ✓ TCAM is very cost-expensive.

Rule Placement Challenges II

- ✓ On receiving a request, for which no flow-rule is present in the switch, the switch sends a *PACKET-IN* message to the controller.
- ✓ The controller decides a suitable flow-rule for the request.
- ✓ The flow-rule is inserted at the switch.
- ✓ Typically, 3-5ms delay is involved in a new rule placement



Rule Placement III

- ✓ How to define/place the rules at switches, while considering available TCAM.
- ✓ How to define rules, so that less number of *PACKET-IN* messages are sent to controller.

OpenFlow Protocol I

- ✓ Only one protocol is available for rule placement – OpenFlow.
- ✓ It has different versions – 1.0, 1.1, 1.2, 1.3, etc. – to have different number of match-fields.

Match SDN Applications First and Use Normal For Unmatched Packets (Hybrid Default Forwarding)

Priority	Ingress Port	MAC Source Address	MAC Destination	Protocol	Vlan ID	IP Source Address	IP Destination	Source Port	Destination Port	Instructions
10000	*	*	*	TCP	*	*	10.1.1.20/32	*	80	Forward to Port 1
5000	*	*	*	*	*	*	10.1.1.0/24	*	*	Forward to Port 2
300	*	*	*	*	2600	*	*	*	*	Send to Controller
0	*	*	*	*	*	*	*	*	*	OF Normal

Source: http://networkstatic.net/wp-content/uploads/2013/06/OFP_normal_rules.png

OpenFlow Protocol II

✓ Different match-fields

- Source IP
- Destination IP
- Source Port
- Priority
- etc.

OpenFlow Protocol III

How much time a flow-rule is to be kept at the switch?

- ✓ Hard timeout
  - All rules are deleted from the switch at hard timeout.
  - This can used to reset the switch.
- ✓ Soft timeout
  - If NO flow is received associated with a rule for a particular time, the rule is deleted.
  - This is used to empty the rule-space by deleting an unused rule.

OpenFlow Protocol IV

- ✓ SDN is NOT OpenFlow
  - SDN is a technology/concept
  - OpenFlow is a protocol used to communicate between data-plane and control-plane.
  - We may have other protocols for this purpose. However, OpenFlow is the only protocol present today.

OpenFlow Switch Software

- ✓ Indigo: Open source, it runs on Mac OS X.
- ✓ LINC: Open source, it runs on Linux, Solaris, Windows, MacOS, and FreeBSD.
- ✓ Pantou: Turns a commercial wireless router/access point to an OpenFlow enabled switch. OpenFlow runs on OpenWRT.
- ✓ Of13softswitch: User-space software switch based on Ericsson TrafficLab 1.1 softswitch.
- ✓ Open vSwitch: Open Source, it is the MOST popular one present today.

Summary

- ✓ Basics of SDN
- ✓ Challenges present in SDN
- ✓ Rule Placement with OpenFlow
- ✓ Controller Placement – to be discussed in next lecture

Thank You!!

Software-Defined Networking – Part II

Restructuring the Current Network Infrastructure

Dr. Sudip Misra
Associate Professor
Department of Computer Science and Engineering
IIT KHARAGPUR
Email: smisra@sit.iitkgp.ernet.in
Website: http://cse.iitkgp.ac.in/~smisra/

SDN - Recap

- ✓ SDN – restructuring current network infrastructure
- ✓ Architecture of SDN – Application, Control and Infrastructure layers
- ✓ Rule Placement – TCAM and Delay
- ✓ OpenFlow protocol – flow-rule and math-fields

APIs in SDN

- ✓ Southbound API
  - Used to communicate between control layer and infrastructure layer.
  - OpenFlow protocol is used.
- ✓ Northbound API
  - Used to communicate between control layer and application layer.
  - Standard APIs are used.
- ✓ East-Westbound APIs
  - Used to communicate among multiple controllers in the control layer.

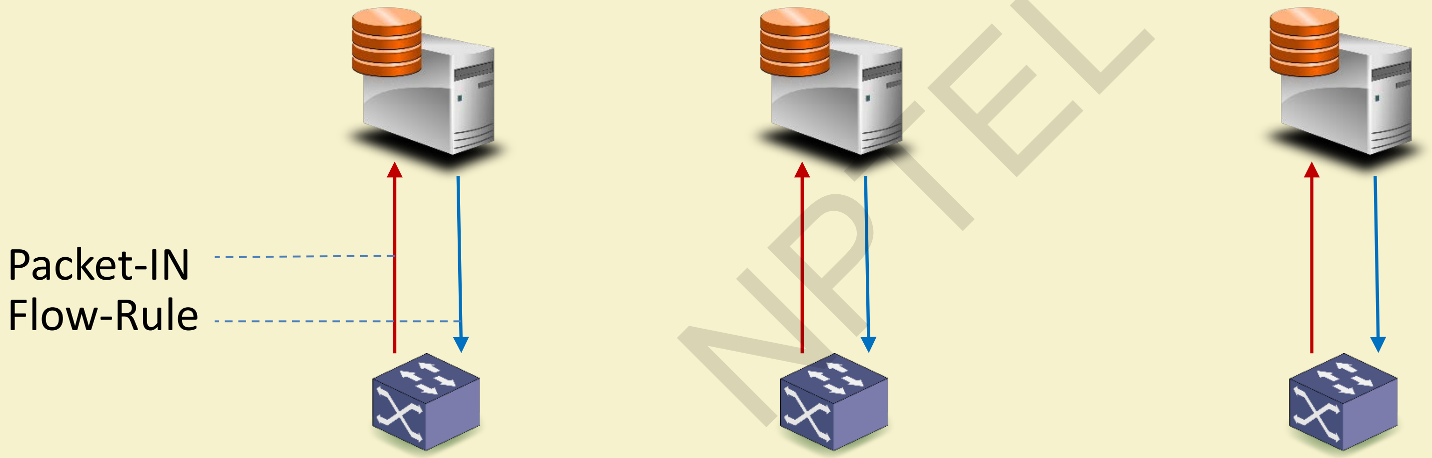
Controller Placement I

- ✓ Controllers define flow-rule according to the application-specific requirements.
- ✓ The controllers must be able to handle all incoming requests from switches.
- ✓ Rule should be placed without incurring much delay.
- ✓ Typically, a controller can handle 200 requests in a second (through a single thread).

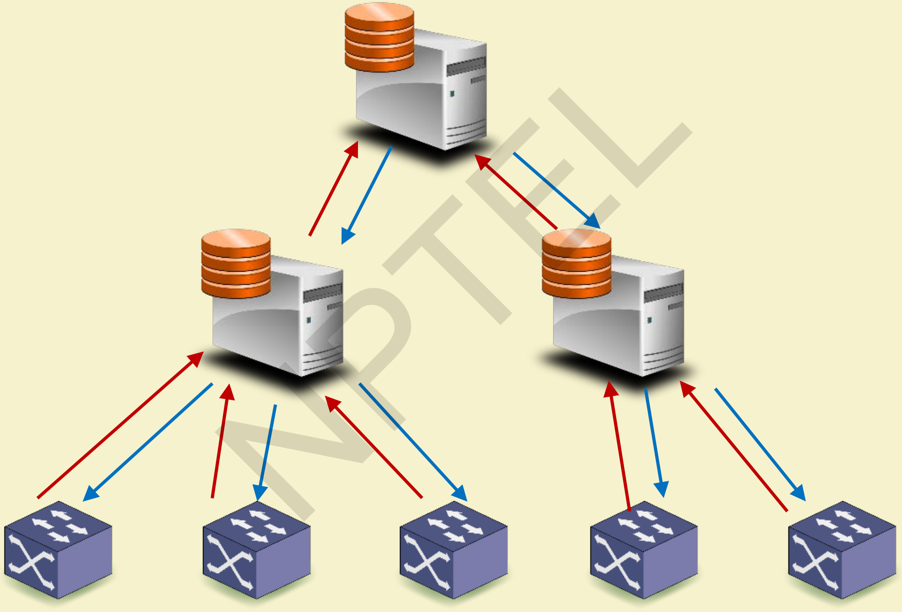
Controller Placement II

- ✓ The controllers are logically connected to the switches in one-hop distance.
  - Physically, they are connected to the switches in multi-hop distance.
- ✓ If we have a very small number of controllers for a large network, the network might be congested with control packets (i.e., PACKET-IN messages).

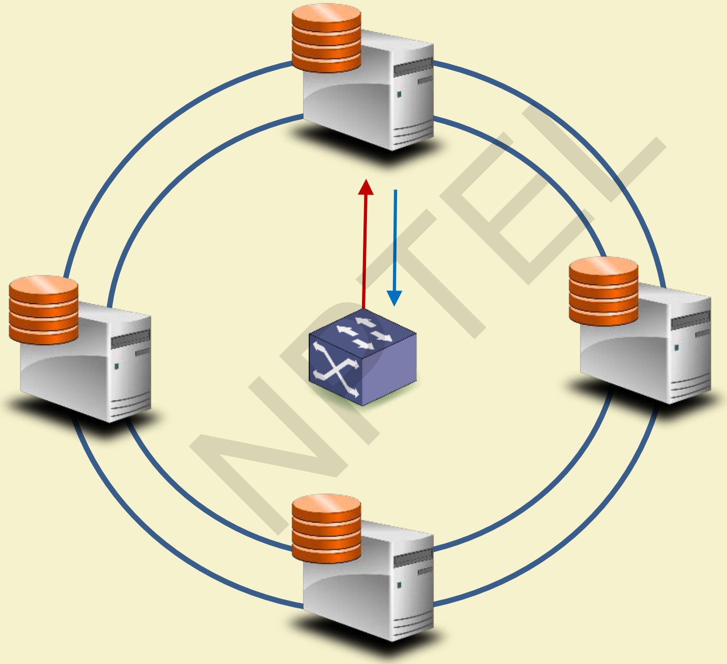
Flat Architecture



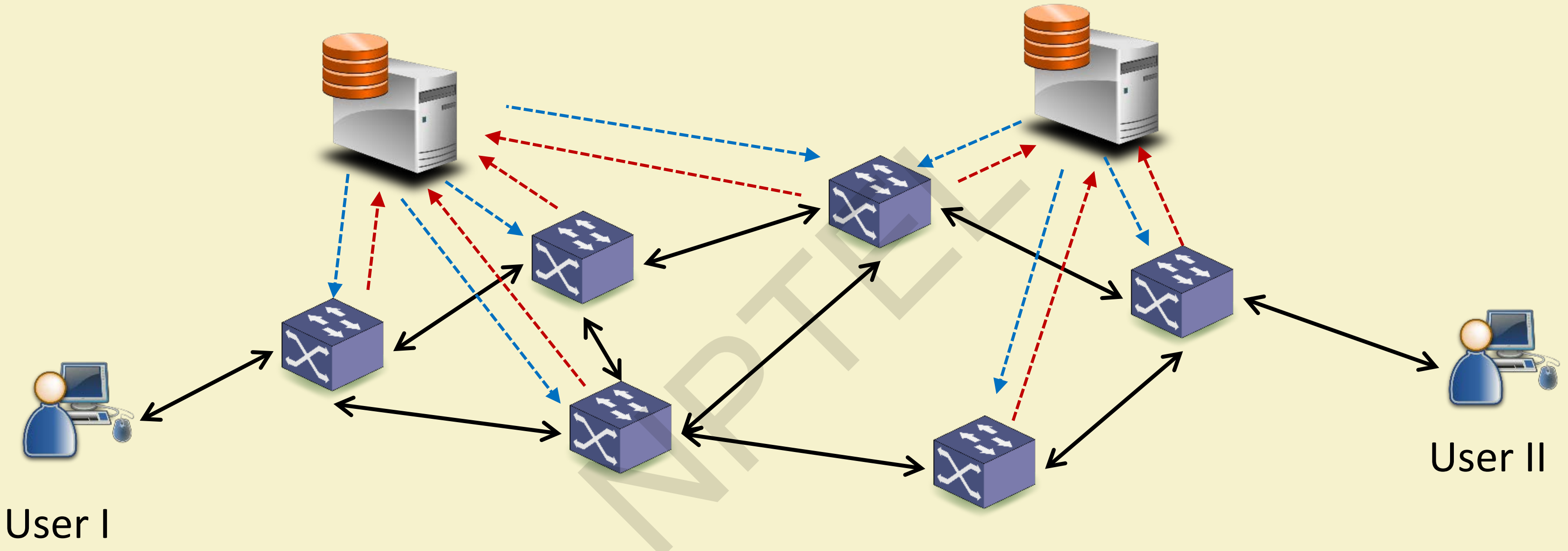
Hierarchical (tree) Architecture



Ring Architecture



Mesh Architecture



Control Mechanisms

- ✓ Distributed
  - The control decisions can be taken in a distributed manner
  - Ex: each subnetwork is controlled by different controller
- ✓ Centralized
  - The control decisions are taken in a centralized manner.
  - Ex: A network is controlled by a single controller.

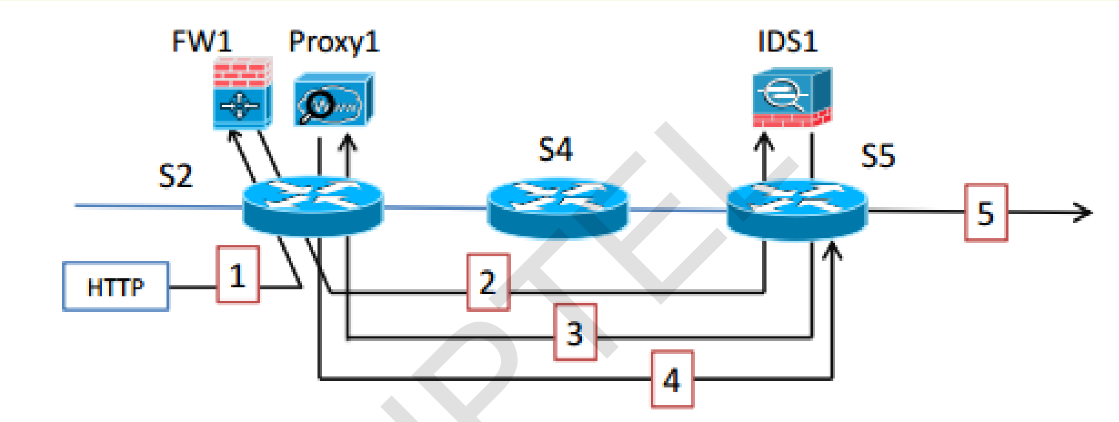
Backup Controller

- ✓ If a controller is down, what will happen?
  - Backup controller is introduced
  - Replica of the main controller is created
  - If the main controller is down, backup controller controls the network to have uninterrupted network management.

Security I

- ✓ Enhanced security using SDN
  - Firewall
  - Proxy
  - HTTP
  - Intrusion detection system (IDS)

Security II



Example of potential data plane ambiguity to implement the policy chain Firewall-IDS-Proxy in the example topology.

Source: SIMPLE-fying Middlebox Policy Enforcement Using SDN, SIGCOMM 2013

Experimenting with SDN

- ✓ Simulator/Emulator
  - Infrastructure deployment – MUST be supported with OpenFlow
  - Controller placement – MUST support OpenFlow
    - Remote – controller can be situated in a remote place, and communicated using IP address and port number
    - Local

Switch Deployment

- ✓ Mininet
  - Used to create a virtual network with OpenFlow-enabled switches
  - Based on Python language
  - Supports remote and local controllers

Controller Configuration Software

- ✓ Pox
- ✓ Nox
- ✓ FloodLight
- ✓ OpenDayLight [Popular!]
- ✓ ONOS [Popular!]

Summary

- ✓ Performance of SDN depends on rule placement and controller placement in the network.
- ✓ Control message overhead may be increased due to additional number of packets (PACKET-IN messages).
- ✓ Unified network management is possible using SDN, while leveraging global view of the network.

Thank You!!

IIT KHARAGPUR

NPTEL ONLINE CERTIFICATION COURSES

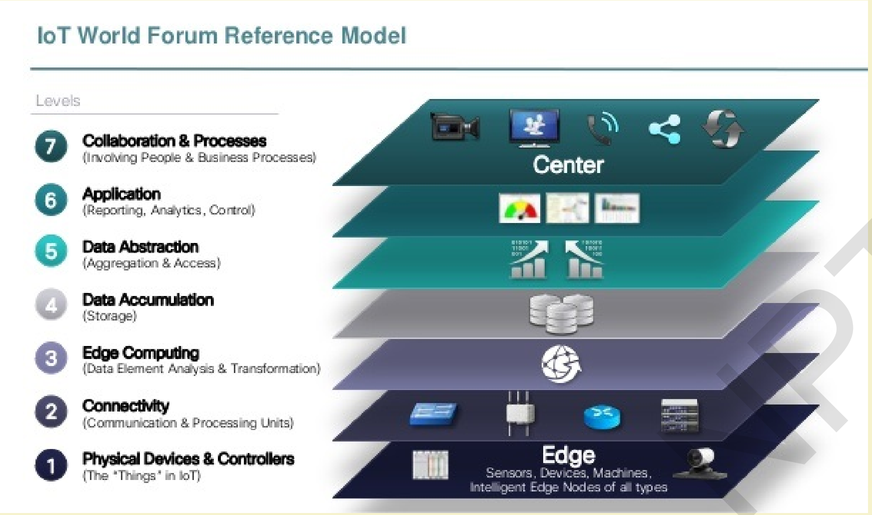
NPTEL

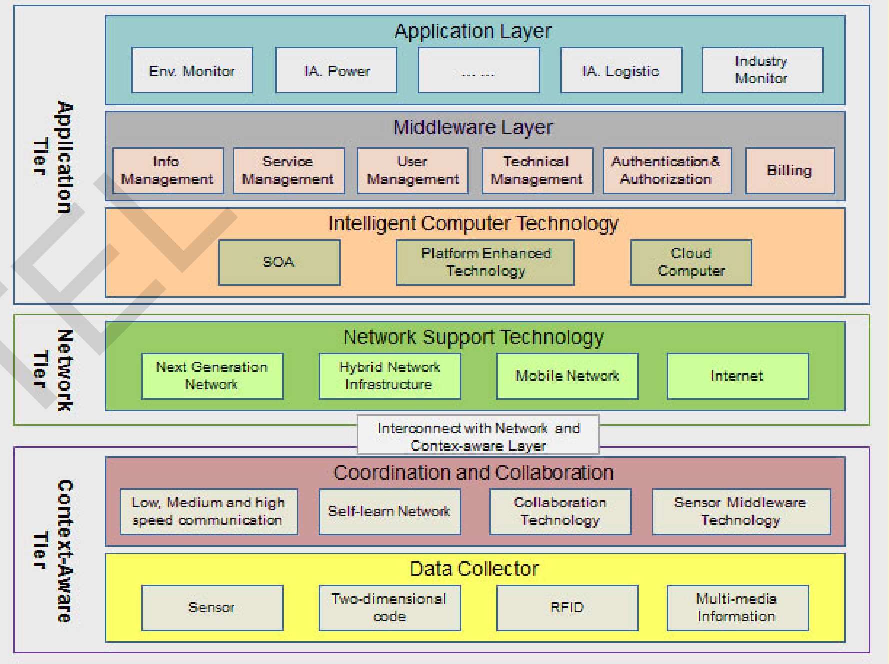
Software-Defined IoT Networking – Part I

Recent Advances of SDN in IoT

Dr. Sudip Misra
Associate Professor
Department of Computer Science and Engineering
IIT KHARAGPUR
Email: smisra@sit.iitkgp.ernet.in
Website: http://cse.iitkgp.ac.in/~smisra/

IoT Architecture





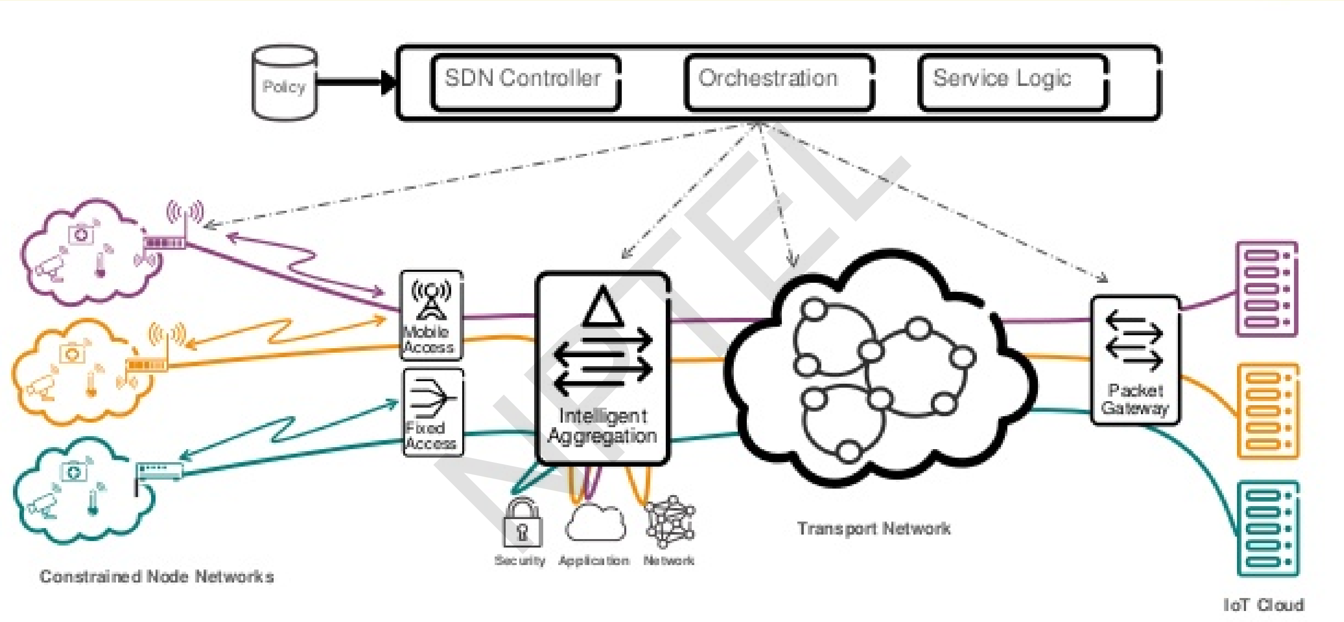
Source: https://image.slidesharecdn.com

Source: http://www.luxhotels.info/p/46800/internet-of-things-iot/

Benefits of Integrating SDN in IoT

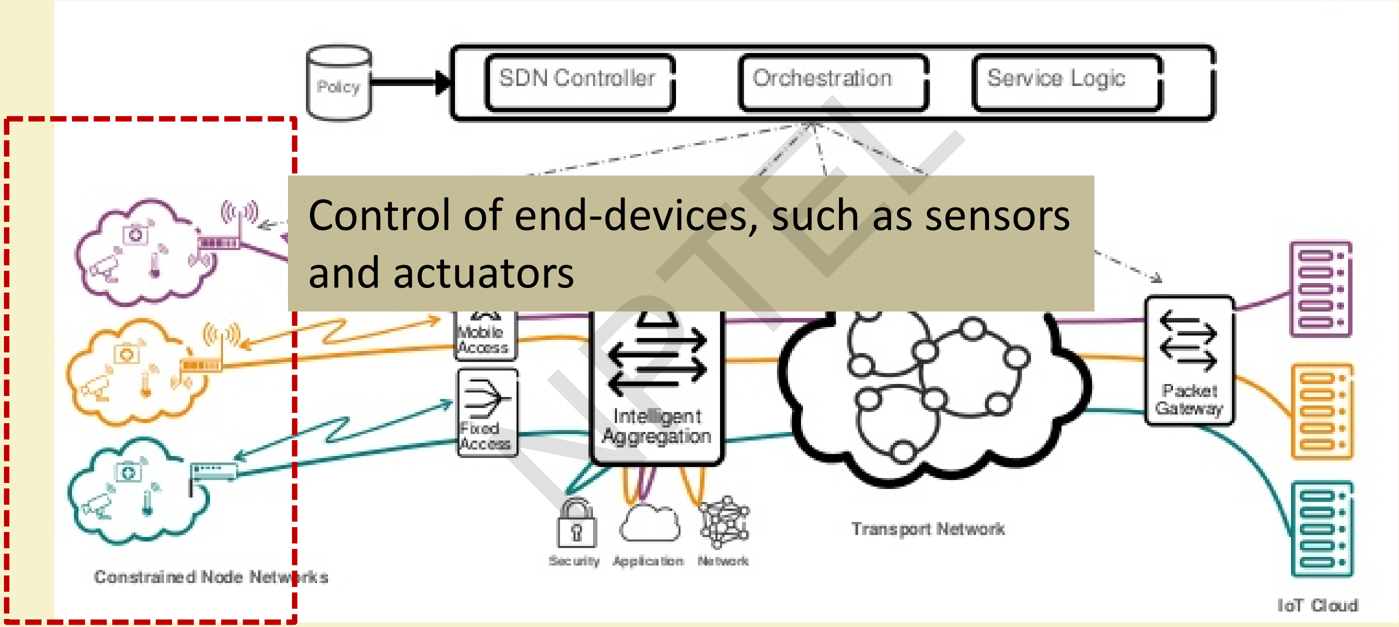
- ✓ Intelligent routing decisions can be deployed using SDN
- ✓ Simplification of information collection, analysis and decision making
- ✓ Visibility of network resources – network management is simplified based on user, device and application-specific requirements
- ✓ Intelligent traffic pattern analysis and coordinated decisions

SDN for IoT I



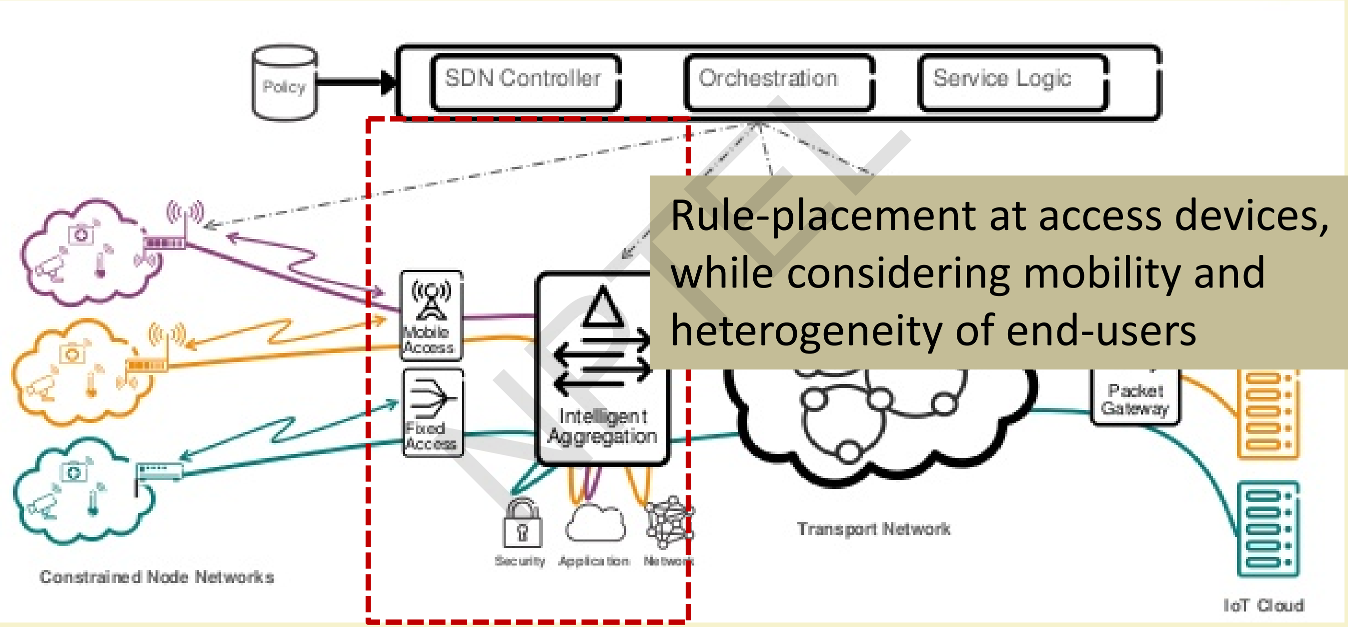
Source: https://image.slidesharecdn.com

SDN for IoT II



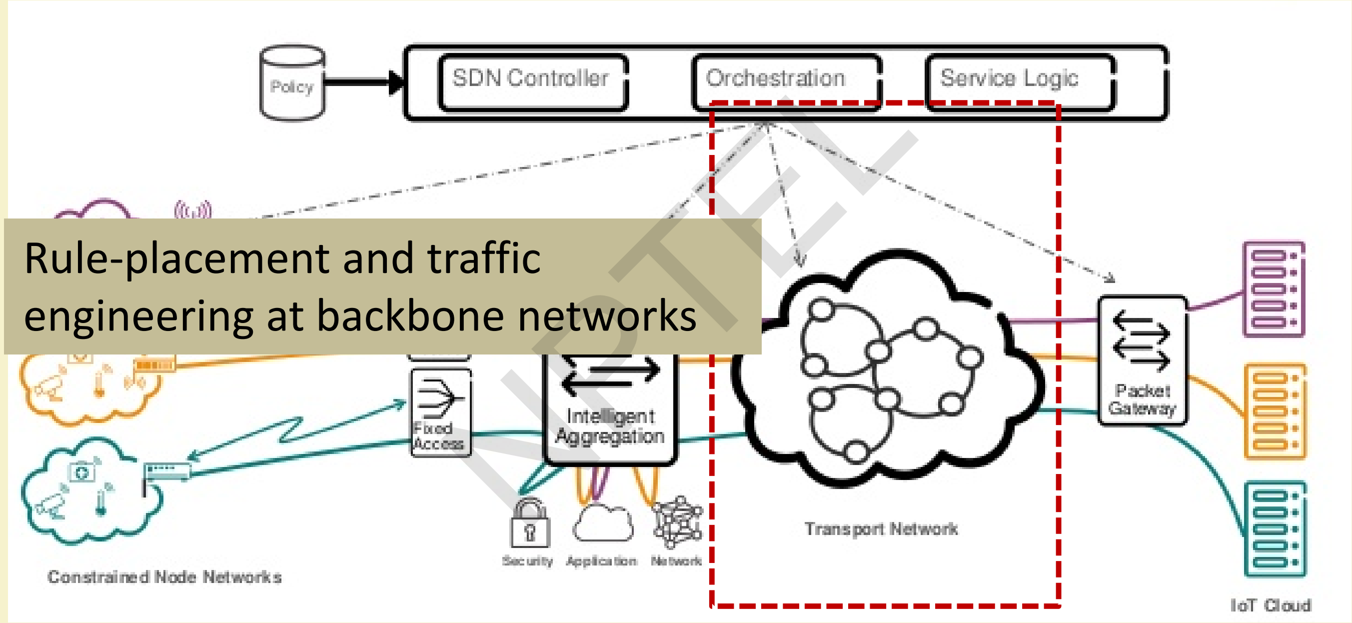
Control of end-devices, such as sensors and actuators

SDN for IoT III



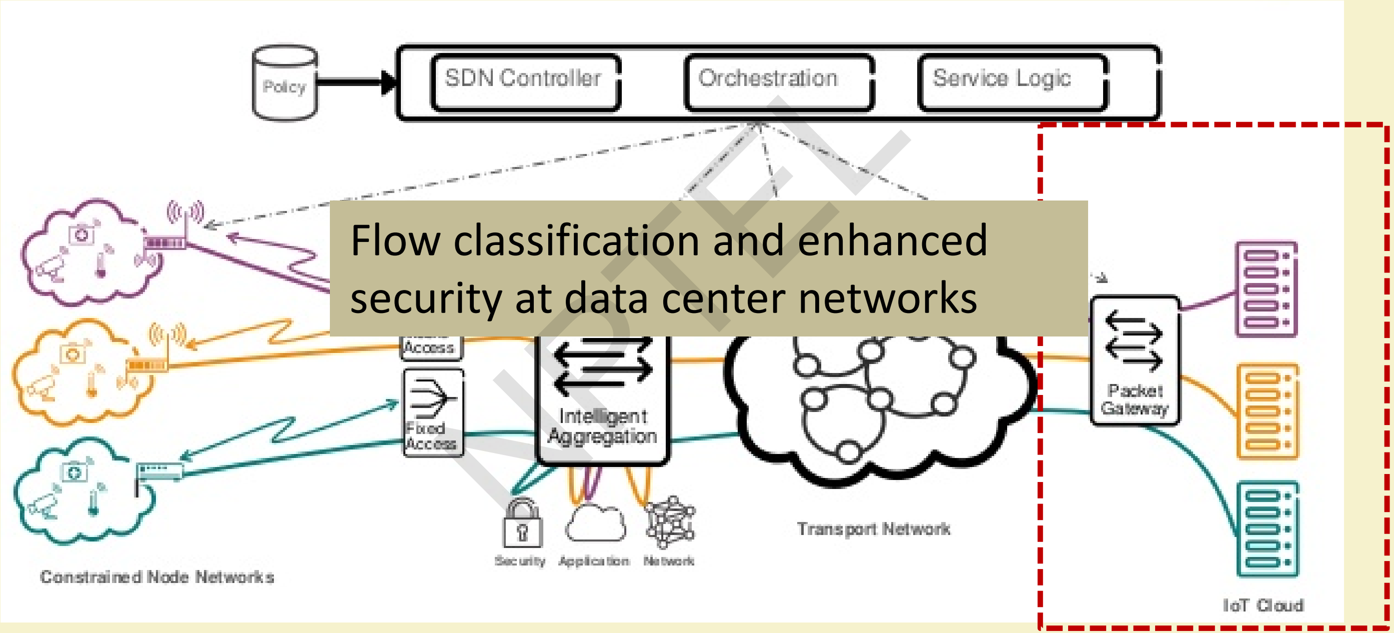
Rule-placement at access devices, while considering mobility and heterogeneity of end-users

SDN for IoT IV



Rule-placement and traffic engineering at backbone networks

SDN for IoT V



Flow classification and enhanced security at data center networks

Wireless Sensor Network I

- ✓ Challenges
  - Real-time programming of sensor nodes
  - Vendor-specific architecture
  - Resource constrained – heavy computation cannot be performed
  - Limited memory – cannot insert too many control programs

Wireless Sensor Network II

✓ Opportunities

- Can we program the sensor nodes in real-time?
- Can we change the forwarding path in real-time?
- Can we integrate different sensor nodes in a WSN?

Software-Defined WSN I

- ✓ Sensor OpenFlow (Luo et al., IEEE Comm. Letters '12)
  - Value-centric data forwarding
    - Forward the sensed data if exceeds a certain value
  - ID-centric data forwarding
    - Forward the sensed data based on the ID of the source node

Real-life implementation of such method NOT done

Software-Defined WSN II

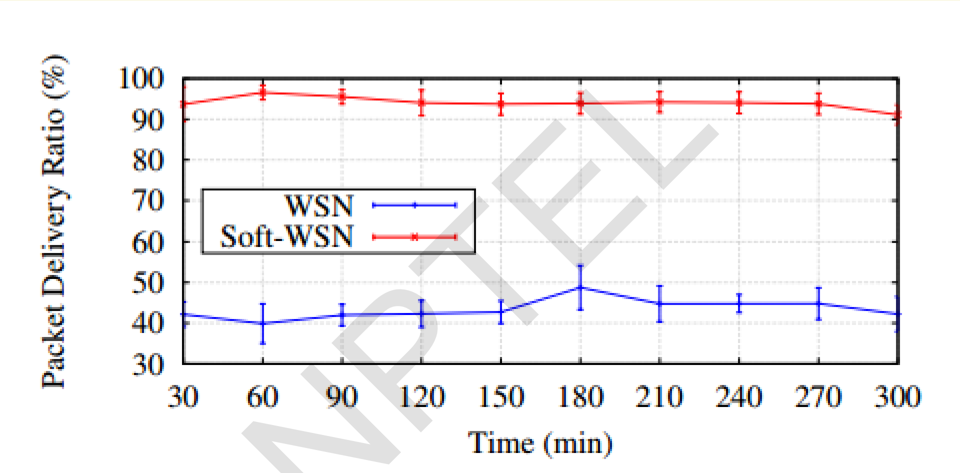
- Soft-WSN (Bera et al., IEEE SJ '16)
  - Sensor Device Management
    - Sensor management
      - Multiple sensors can be implemented in a single sensor board
      - Sensors can be used depending on application-specific requirements
    - Delay management
      - Delay for sensing can be changed dynamically in real-time
    - Active-Sleep Management
      - States of active and sleep mode can be changed dynamically

Software-Defined WSN III

- ✓ Soft-WSN
  - Topology Management
    - Node-specific management – forwarding logic of a particular sensor can be modified
    - Network-specific management
      - Forward all traffic of a node in the network
      - Drop all traffic of a node in the network

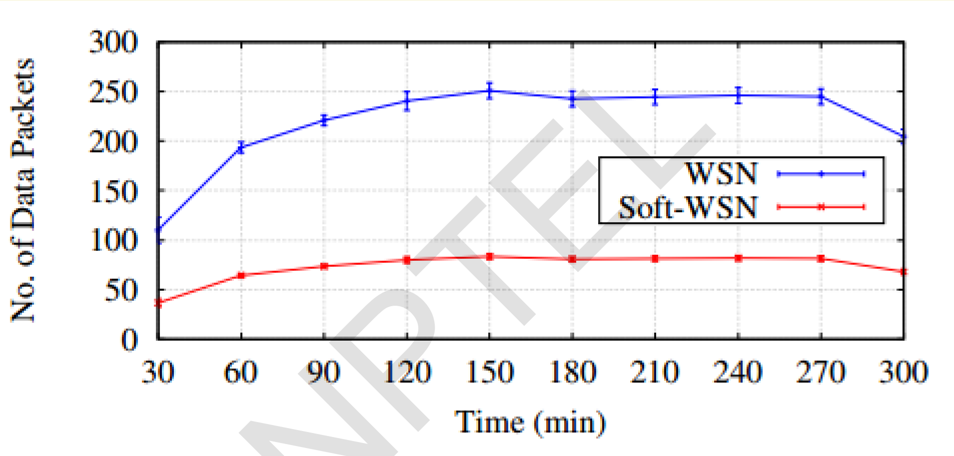
Experimental results show that network performance can be improved using software-defined WSN over traditional WSN

Soft-WSN: Result I



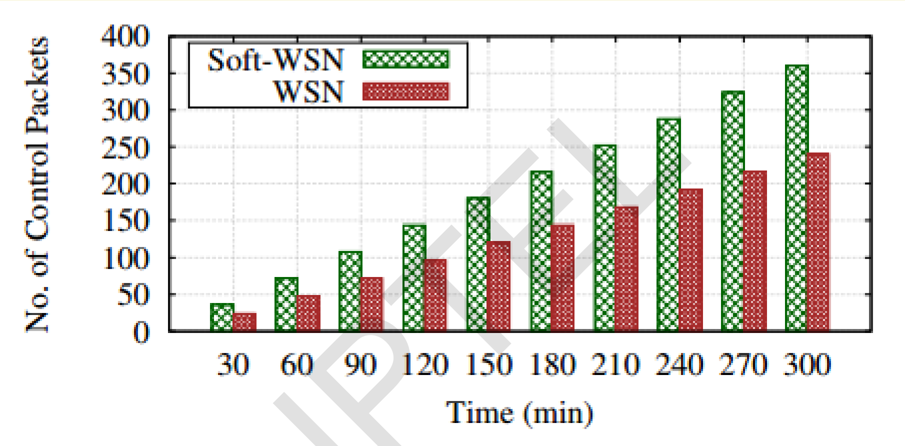
Packet delivery ratio in the network increases using Soft-WSN compared to the traditional WSN.

Soft-WSN: Result II



Number of replicated data packets is reduced using Soft-WSN over the traditional WSN.

Soft-WSN: Result III



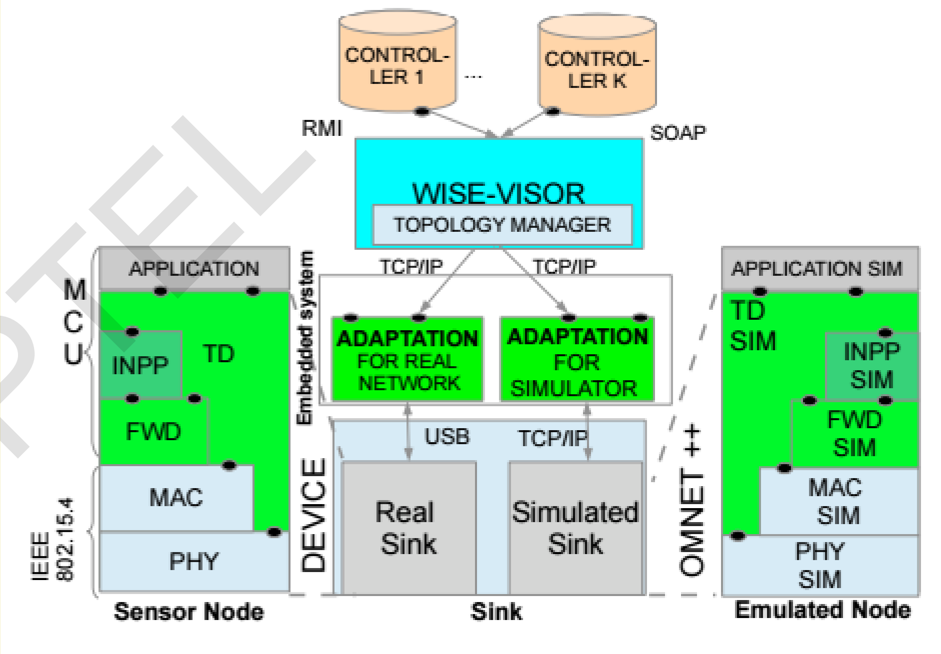
Number of control messages in the network is higher using Soft-WSN over the traditional WSN. This is due to the PACKET-IN message in the network. Each time a node receives a new packet, it asks the controller for getting adequate forwarding logic.

Software-Defined WSN III

- ✓ SDN-WISE (Galluccio et al., IEEE INFOCOM ’15)
  - A software-defined WSN platform is designed
  - Flow-table for rule placement at sensor nodes is designed
  - Any programming language can be used through API to program the nodes in real-time

SDN-WISE Protocol Stack

- ✓ Sensor node includes
  - IEEE 802.15.4 protocol
  - Micro control unit (MCU)
  - Above IEEE 802.15.4 stack, *Forwarding* layer consists of *Flow-rules.*
  - *INPP* – In Network Packet Processing



Source: Galluccio et al., IEEE INFOCOM ’15

Summary

- ✓ SDN is useful to manage and control IoT network
- ✓ Wireless sensor nodes and network can be controlled using SDN-based applications
- ✓ Network performance can be improved significantly using SDN-based approaches over the traditional approaches

Thank You!!