Chapter 6
Introduction to Modern Networking Paradigms

Dr. Nilesh Patil

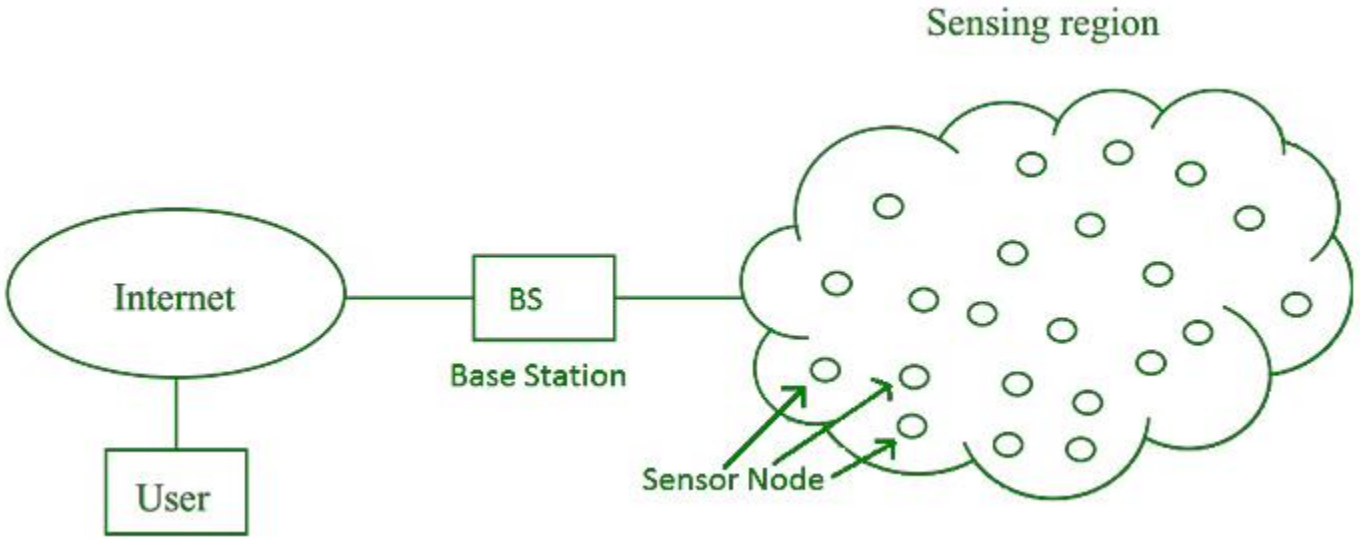
Associate Professor, DJSCE

Unit	Description	Duration	CO
6	Introduction to Modern Networking Paradigms: Introduction to WSN, WSN Architecture, Types of WSN, Challenges in WSN. Introduction to SDN, SDN Architecture, Key Concepts in SDN (Introduction to flow-based forwarding and an overview of the OpenFlow protocol.)	03	CO6

What is a Wireless Sensor Network (WSN)?

- ***Wireless Sensor Network (WSN) is a self-configured, infrastructure-less wireless network.***
- ***WSN network comprises a group of wireless sensor*** *nodes* ***that communicate wirelessly and are distributed in an ad-hoc manner (randomly) to monitor various conditions, such as environmental or physical parameters within a system.***
- In a WSN, each sensor node is a small but powerful device equipped with a microcontroller, radio frequency receiver and transceiver, power source, and memory for wireless communication.
- These nodes are designed to operate independently, configuring themselves into a network without needing a pre-existing infrastructure or transmission media, such as cables.
- Sensor nodes can collect data continuously or in response to specific events, like a security camera that only records when it detects movement.
- The data collected by individual sensor nodes is transmitted to a central node known as the Base Station in a WSN System.
- The Base Station acts as a point or place where data from across the network is compiled and sent through the Internet.

Wireless Sensor Network



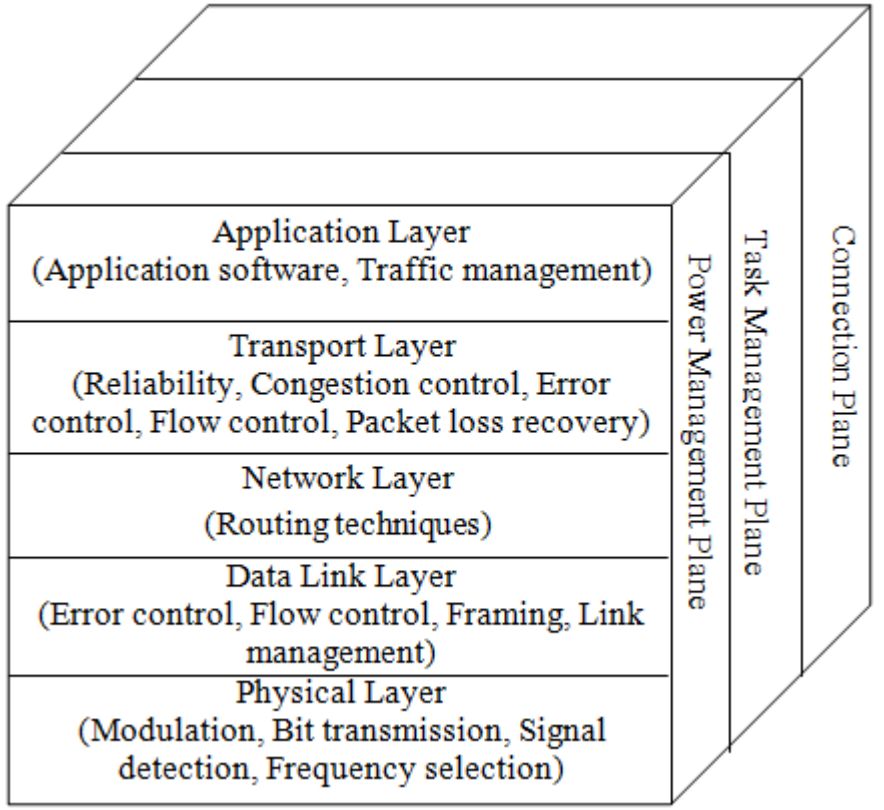
WSN Architecture

- WSNs typically follow a layered architecture, similar to the OSI model, to facilitate efficient communication and management.
- Key layers include:

1. Application Layer
2. Transport Layer
3. Network Layer
4. Data Link Layer
5. Physical Layer

- The three cross layers include the following:

1. Power Management Plane
2. Mobility Management Plane
3. Task Management Plane



- The protocol stack of wireless sensor network consists of five layers: physical layer, data link layer, network layer, transport layer and application layer as shown in Figure.

1. Physical layer provides facilities of modulation, transmission and receiving techniques.
2. Data link layer provides services such as medium access, data transmission, flow control and error control.
3. Network layer provides the facility of routing the data provided by the transport layer.
4. Transport layer protocols provide services such as reliability, packet loss recovery, congestion control, flow control, energy efficiency and heterogeneous application support.
5. Application layer consists of various protocols which provide numerous sensor network services.

- The protocol stack for wireless sensor network can also be divided into three management planes across each layer. The planes are task management, power management and connection management.

1. The task management plane distributes tasks among sensor nodes to provide energy efficiency and increase network lifetime.
2. The power management plane deals with the power level of sensor nodes for sensing, processing, transmission and reception of data.
3. The connection management plane maintains the network connectivity in case of situations like topology changes and node deployment.

Types of WSN

- WSNs can be categorized based on various factors, including deployment environment and application. Some common types include:

1. **Terrestrial WSNs:** These networks are deployed on land and are used for applications like environmental monitoring, agriculture, and infrastructure monitoring.
2. **Underground WSNs:** These networks are deployed underground and are used for applications like mining, tunneling, and underground infrastructure monitoring.
3. **Underwater WSNs:** These networks are deployed underwater and are used for applications like oceanographic research, underwater surveillance, and pipeline monitoring.
4. **Multimedia WSNs:** These networks handle multimedia data, such as images and videos, and are used for applications like surveillance and remote sensing.
5. **Mobile WSNs:** These networks consist of mobile sensor nodes and are used for applications like tracking, disaster relief, and military operations.

Applications of Wireless Sensor Network

- WSN is introduced in the following applications:

1. Battlefield – Surveillance and monitoring the movement of enemies and own military forces.
2. Disaster relief operation – receiving data and analyzing situation of the affected area.
3. Environmental use – monitoring different environmental parameters of a particular region like temperature, air pressure, rain etc.; tracking related data of flora and fauna of certain biosphere.
4. Agriculture – monitoring data regarding soil, weather, irrigation for agricultural use.
5. Health care – monitoring patients' physical condition and giving necessary feedback in alarming situation.
6. IoT – The Internet of Things works on the basis of physical world of devices and objects connected over the network using the wireless sensors.

Challenges of WSN

- In spite of their highly practical usefulness there are some challenges in wireless sensor network system –.

1. Scalability – There are a vast difference in scale of such sensor networking system as the number of sensor nodes may vary from few to several. Added to this the deployment density is correspondingly adjustable.
2. Energy efficiency – As wireless sensor nodes have to work on a limited power supply, the designing of the software and the hardware has to be so optimized that it can perform efficiently the designated job.
3. Maintenance – WSN has several constraints like power supply, storage, large amount of algorithms, so there is a serious challenge in maintenance of all these.
4. Security – Like all internet dependent applications, WSN also has insecurity scare. Proper data transmission management should be adopted to counter data theft by every possible way.
5. Quality of service – The data must be provided in time as the real time based applications heavily dependent on the timely distributed data.

Software-Defined Networking (SDN): Introduction

- Software-defined networking (SDN) is a software-controlled approach to networking architecture driven by application programming interfaces (APIs).
- SDN leverages a centralized platform to communicate with IT infrastructure and direct network traffic.
- SDN is an architecture designed to make a network more flexible and easier to manage.
- SDN centralizes management by abstracting the control plane from the data forwarding function in the discrete networking devices.

Why Software Defined Networking is Important?

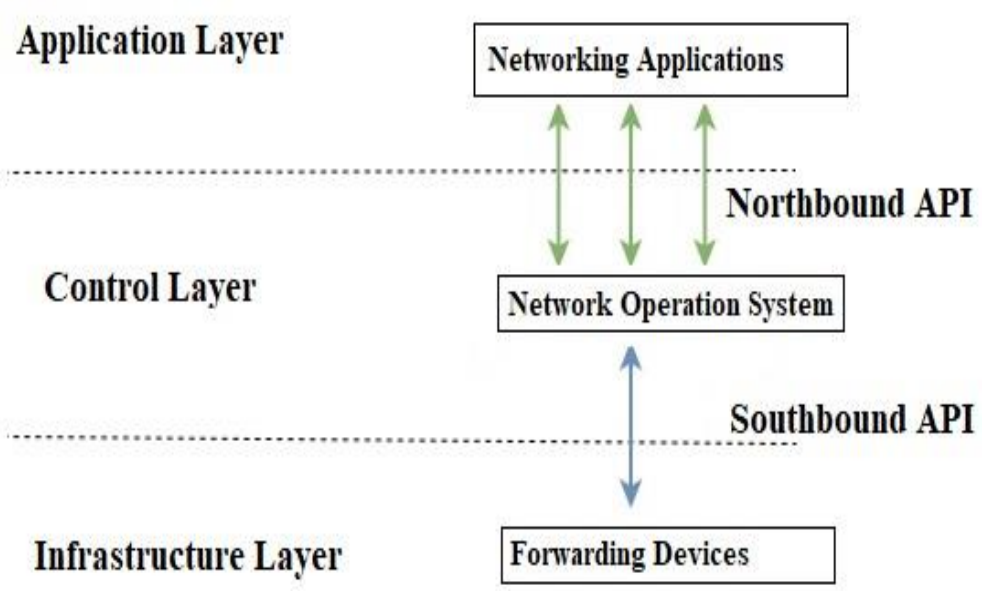
- **Better Network Connectivity:** SDN provides very better network connectivity for sales, services, and internal communications. SDN also helps in faster data sharing.
- **Better Deployment of Applications:** Deployment of new applications, services, and many business models can be speed up using Software Defined Networking.
- **Better Security:** Software-defined network provides better visibility throughout the network. Operators can create separate zones for devices that require different levels of security. SDN networks give more freedom to operators.
- **Better Control With High Speed:** Software-defined networking provides better speed than other networking types by applying an open standard software-based controller.

Elements of SDN

- An SDN architecture delivers a centralized, programmable network and consists of the following:

1. A controller, the core element of an SDN architecture, that enables centralized management and control, automation, and policy enforcement across physical and virtual network environments
2. Southbound APIs that relay information between the controller and the individual network devices (such as switches, access points, routers, and firewalls)
3. Northbound APIs that relay information between the controller and the applications and policy engines, to which an SDN looks like a single logical network device

SDN Architecture

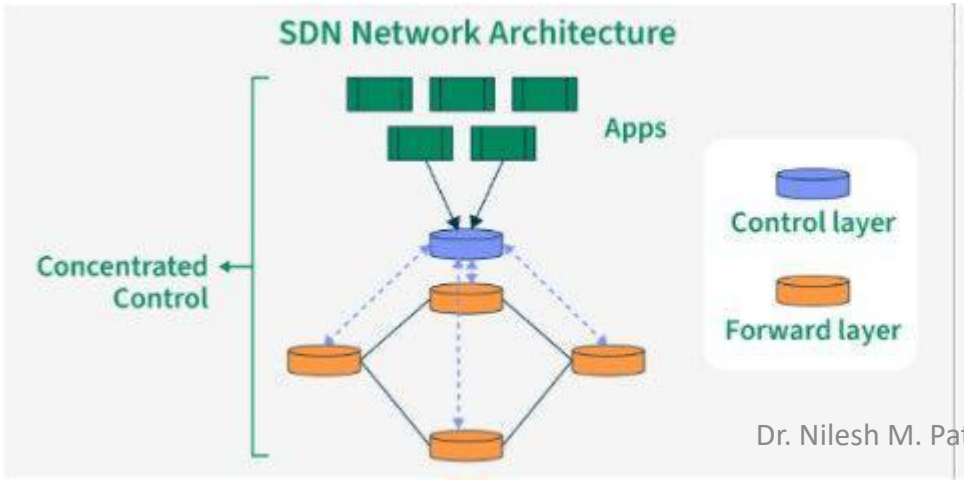


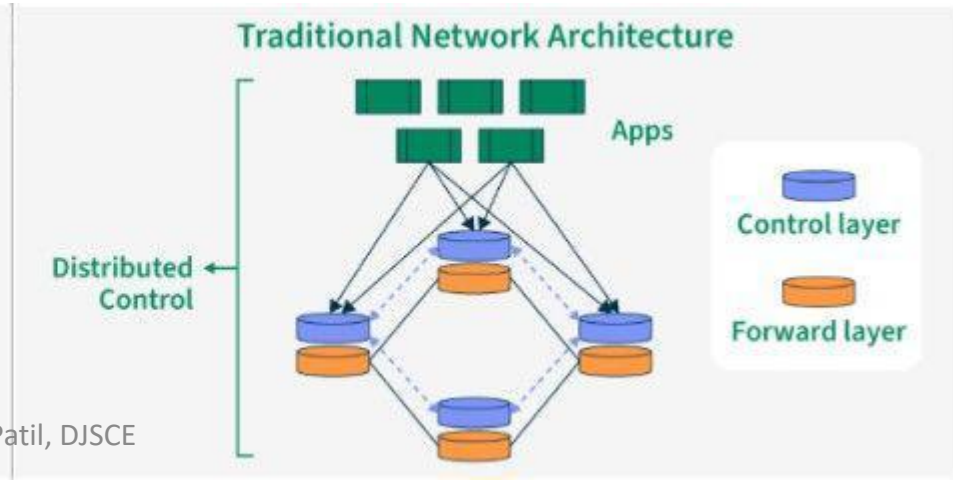
- SDN architecture consists of three major layers:

1. **Application Layer:** It is responsible for developing and managing networking applications. Application layer deals with end-user business applications that utilizes the SDN services. Business application such as energy efficient networking, security monitoring, network virtualization etc.
2. **Control Layer:** The control layer is also referred as control plane that comprises a set of software-enabled SDN controllers. This layer allows the network administrator to apply custom policies to the physical layer devices. It is responsible for routing packets to their respective destination while applying different rules and policies.
3. **Infrastructure Layer:** This layer consists of forwarding devices like the physical switch, router, etc. Software switches which can be accessible via open interfaces, also part of this layer. This layer is considered as forwarding layer since it allows packet switching and forwarding.

Difference Between SDN and Traditional Networking

Software Defined Networking	Traditional Networking
Software Defined Network is a virtual networking approach.	A traditional network is the old conventional networking approach.
Software Defined Network is centralized control.	Traditional Network is distributed control.
This network is programmable.	This network is nonprogrammable.
Software Defined Network is the open interface.	A traditional network is a closed interface.
In Software Defined Network data plane and control, the plane is decoupled by software.	In a traditional network data plane and control plane are mounted on the same plane.





Advantages of SDN

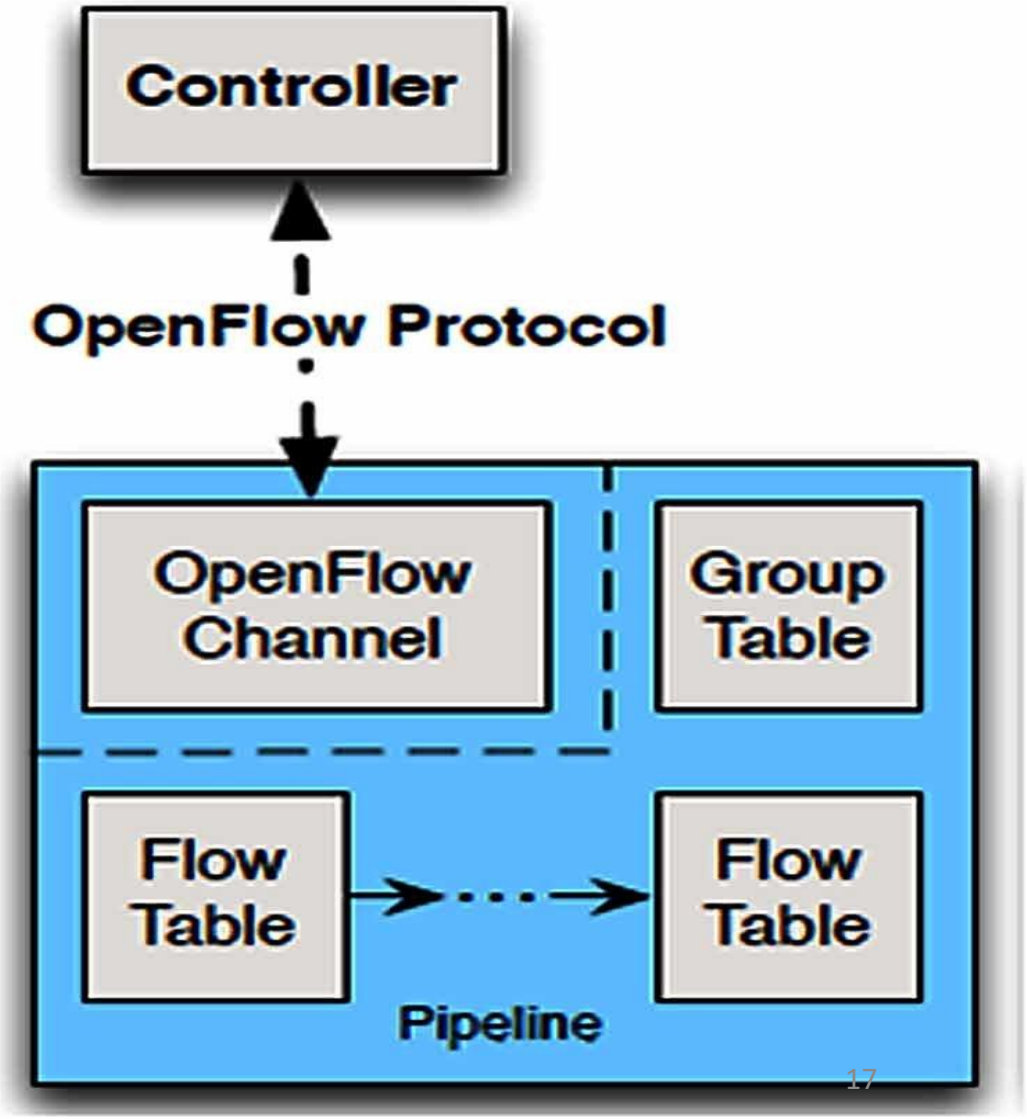
- The network is programmable and hence can easily be modified via the controller rather than individual switches.
- Switch hardware becomes cheaper since each switch only needs a data plane.
- Hardware is abstracted; hence applications can be written on top of the controller independent of the switch vendor.
- Provides better security since the controller can monitor traffic and deploy security policies. For example, if the controller detects suspicious activity in network traffic, it can reroute or drop the packets.

Disadvantages of SDN

- The central dependency of the network means a single point of failure, i.e. if the controller gets corrupted, the entire network will be affected.
- The use of SDN on large scale is not properly defined and explored.

OpenFlow in SDN

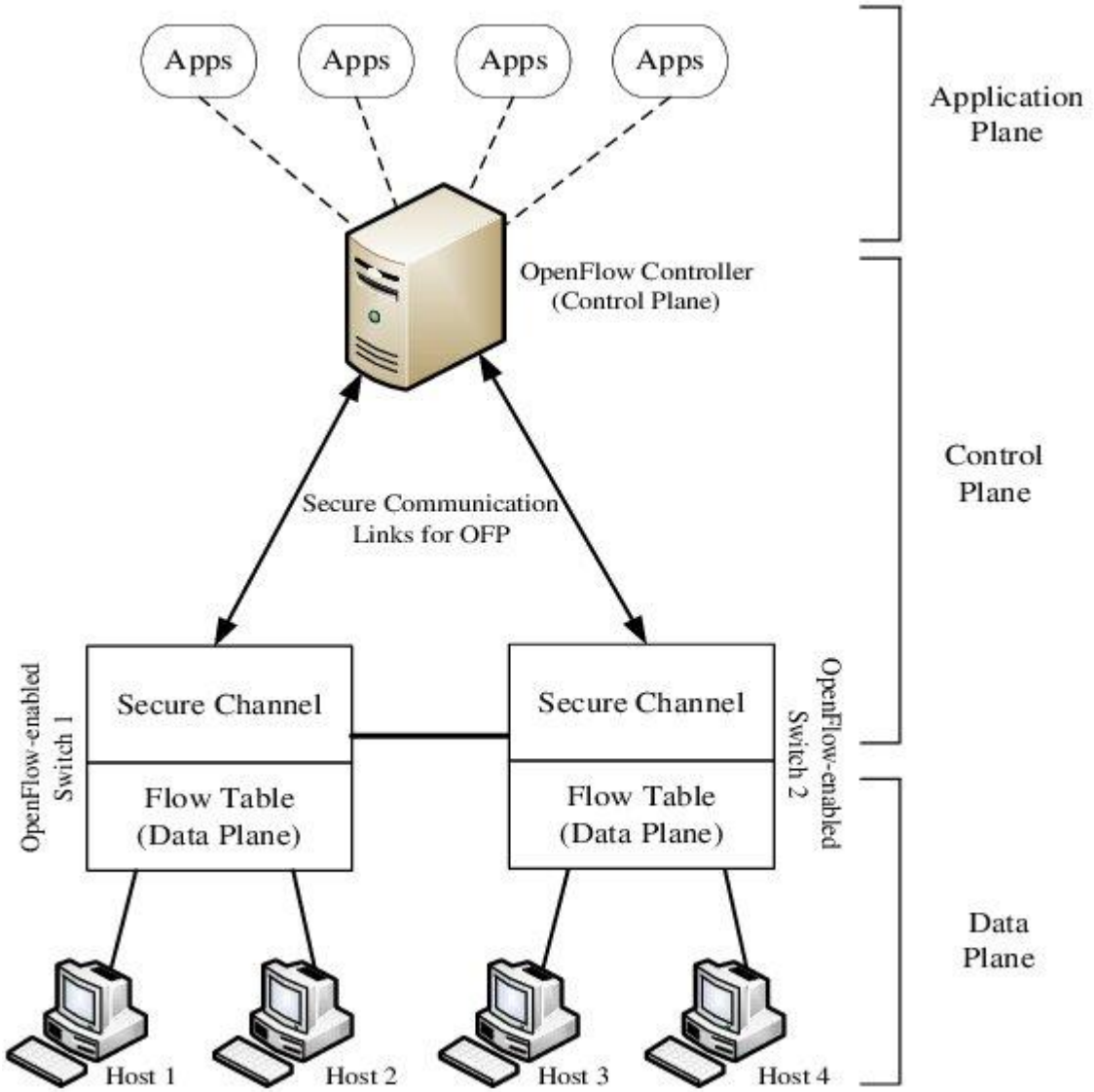
- It is a multivendor standard defined by the Open Networking Foundation (ONF) for implementing SDN in networking equipment.
- The OpenFlow protocol defines the interface between an OpenFlow Controller and an OpenFlow switch as shown in figure .
- The OpenFlow protocol allows the OpenFlow Controller to instruct the OpenFlow switch on how to handle incoming data packets.



OpenFlow Switch

- The OpenFlow switch may be programmed to:
  (1) identify and categorize packets from an ingress port based on a various packet header fields;
  (2) Process the packets in various ways, including modifying the header; and,
  (3) Drop or push the packets to a particular egress port or to the OpenFlow Controller.
- The OpenFlow instructions transmitted from an OpenFlow Controller to an OpenFlow switch are structured as “flows”.
- Each individual flow contains packet match fields, flow priority, various counters, packet processing instructions, flow timeouts and a cookie.
- The flows are organized in tables.
- An incoming packet may be processed by flows in multiple “pipelined” tables before exiting on an egress port.

OpenFlow Network Architecture (1/3)



- Unlike traditional networking architecture, in OpenFlow-enabled networks control plane and data plane are separated from network core devices as discussed above.
- The control plane resides on top of all data planes, which is a central controller. Thus, OpenFlow network architecture is referred as centrally control architecture.
- Software running inside OpenFlow controller is called NOS which act as an intermediate plane between data plane and application plane.
- The OpenFlow network architecture consists of three layers:
    1. Lowest layer is data forwarding plane and it includes one or more OpenFlow-enabled virtual or physical switches.
    2. Second layer is a control plane and it includes OpenFlow controllers with predefined NOS. Sometimes one or more controller may also require for complex network design.
    3. The third and topmost layer is an application plane. One or more OpenFlow application is defined at this layer for management or data flows controlling task.

OpenFlow Network Architecture (2/3)

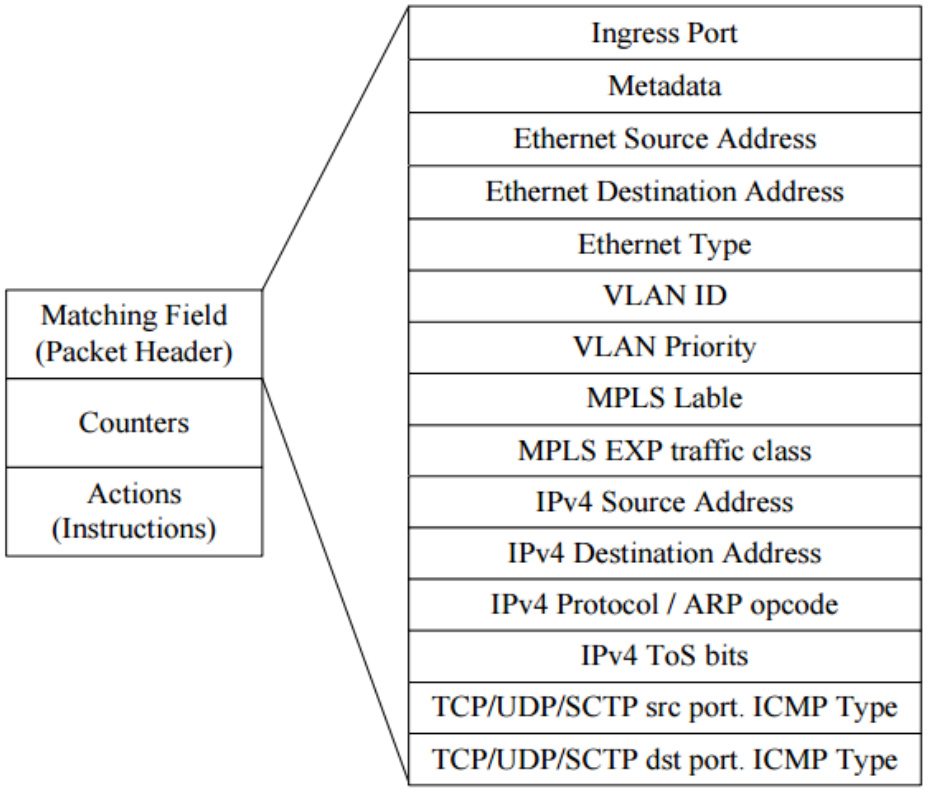
- In an architecture shown, three layers are stacked starting from lowest data plane, control plane and application plane.
- Data plane comprises of network core devices for forwarding data packets, these devices are having flow tables (or look-up tables).
- Control Plane is a NOS running in OpenFlow controller and Application Plane defines various management or control applications.
- Whenever any data packet from end host arrives at an OpenFlow-enabled switch, the switch will forward this packet to a control plane for verification.
- The function of switch is to encapsulate and forwards the first packet arrives from end host to an OpenFlow controller on secure link using OpenFlow Protocols (OFP). This in turn enables the controller to decide whether the flow should be added to flow table of switches or to discard.
- OpenFlow switch consists of flow table and secure channel to communicate with OpenFlow controller using OpenFlow Protocols (OFP).
- Each data flow through the network must first get permission from the OpenFlow controller in order to verify whether communication is permissible by network policies or not.
- If controller allows the flow than it will compute the route and inserts the flow entries in the flow table of an OpenFlow switch. The flow table entries done by controller have three fields.
- Once an entry is done by controller in a switch, all the succeeding packet arrives from hosts to a switch will match the entry and follow the same path dictated by a controller.
- If entry is not found in the flow table, then either switch will discard the packet, or it will send to the controller for further processing based on controller decision.

OpenFlow Network Architecture (3/3)

- The three fields of Flow Table are:

1. Matching Field (or Packet Header): This field is used to match the updated information in a flow table against the arrived data packet. It consists of ingress port and other header fields.
2. Counters: This field is used to update the statistics of a packet after matching has done and to keep track on the number of packets and number of bytes for each flows and also to maintain the timing constraints.
3. Actions: This field is used to perform the specific actions set up by a controller to process an incoming packet after matching has done.

- The flow entries in a flow table of a switch are removed after some timeout time called idle timeout.
- The flow entries can also be added or removed manually through some sort of software or hardware constraints.



THANK YOU