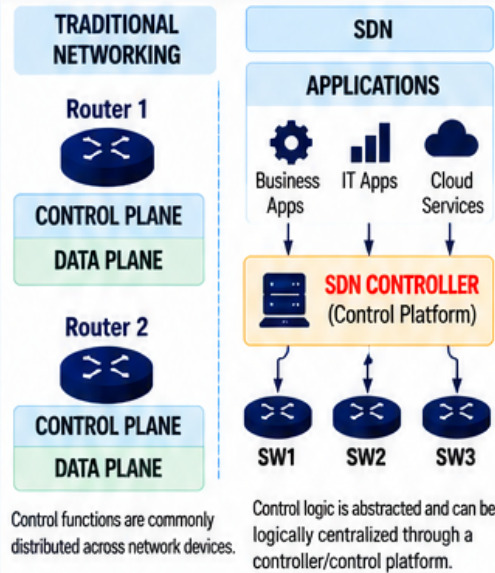


SDN QUICK FACTS

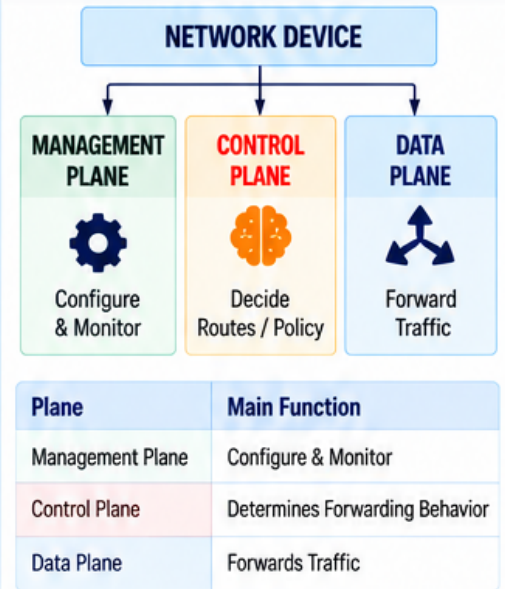
Concept	SDN
 Full Name	Software-Defined Networking
 Main Idea	Programmable Network Control
 Control	Logically Centralized
 Forwarding	Network Infrastructure
 Key Component	Controller / Control Platform
 Goal	Automation & Programmability

**SDN = PROGRAMMABLE CONTROL
OF NETWORK BEHAVIOR**

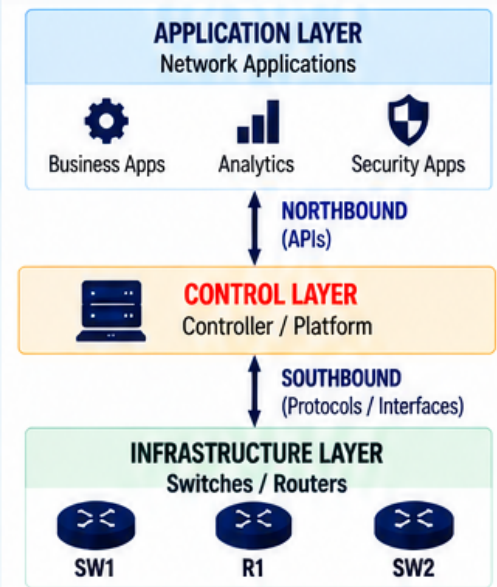
TRADITIONAL NETWORKING vs SDN



THREE NETWORK PLANES

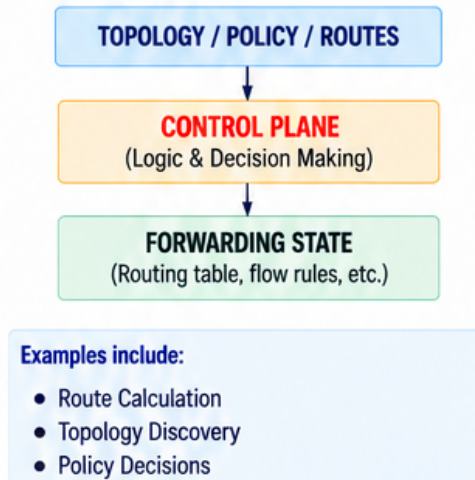


SDN ARCHITECTURE



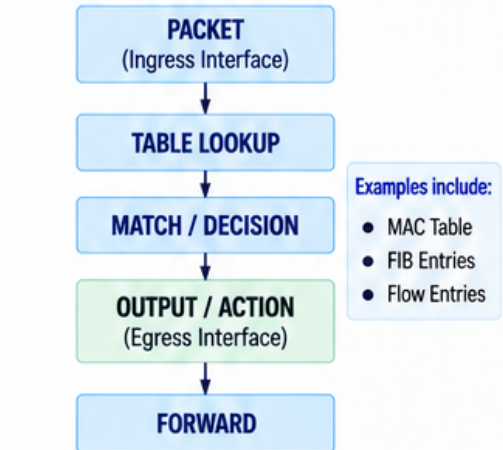
CONTROL PLANE

The control plane determines how network traffic should be forwarded.



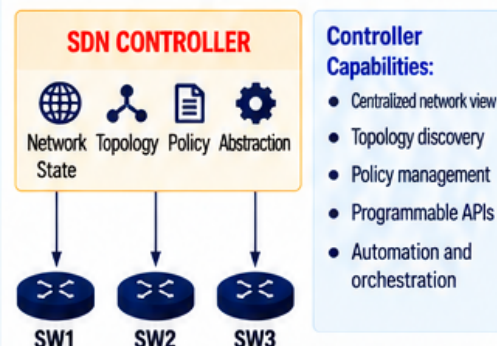
DATA PLANE

The data plane, or forwarding plane, forwards traffic according to installed forwarding state.

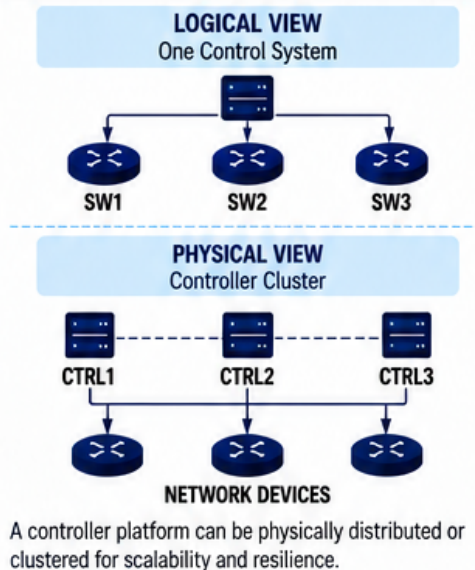


SDN CONTROLLER

The controller/control platform provides a programmable abstraction and control point for network resources.



LOGICALLY vs PHYSICALLY CENTRALIZED



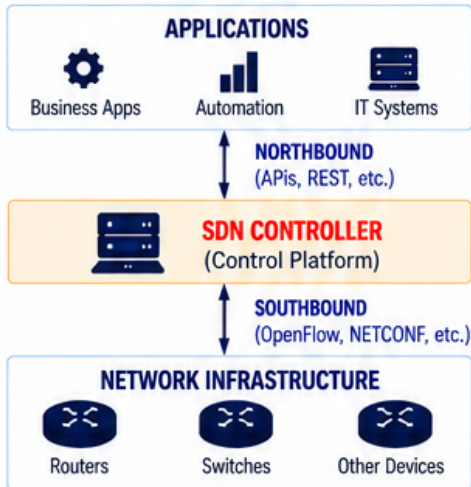
WHY SDN?

Traditional Approach	SDN Approach
Device-Centric	Software / Policy-Driven
Device-by-Device Changes Common	Centralized Policy Possible
CLI Workflows Common	API-Based Programmability
Distributed Device Control	Logically Centralized Control
Manual Operations Common	Automation-Friendly

Key Benefits of SDN:

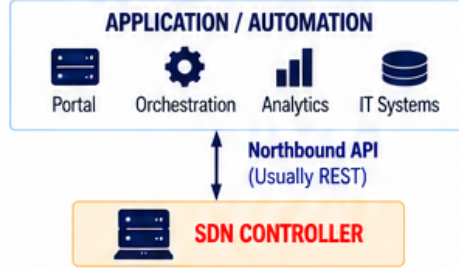
- ✓ Programmability
- ✓ Automation
- ✓ Abstraction
- ✓ Consistency
- ✓ Faster Service Deployment

SDN COMMUNICATION MODEL



NORTHBOUND APIs

Northbound APIs expose controller capabilities and network abstractions to applications and automation systems.

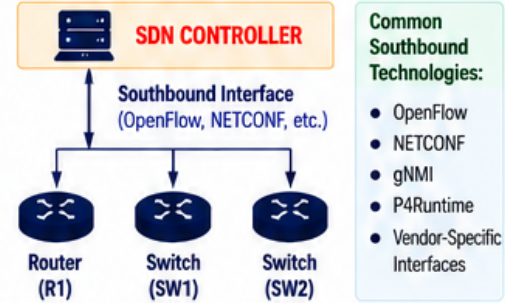


Typical uses:

- Network state and topology
- Policy and intent
- Service requests
- Automation and orchestration

SOUTHBOUND INTERFACES

Southbound interfaces enable communication between the controller/control platform and network infrastructure.



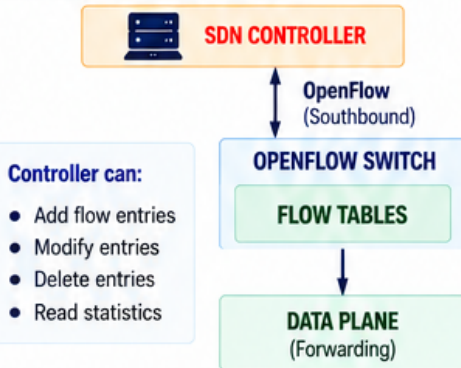
Common Southbound Technologies:

- OpenFlow
- NETCONF
- gNMI
- P4Runtime
- Vendor-Specific Interfaces

The exact southbound technology depends on the SDN implementation.

OPENFLOW

OpenFlow is a well-known southbound protocol historically associated with SDN.



Controller can:

- Add flow entries
- Modify entries
- Delete entries
- Read statistics

OPENFLOW FLOW ENTRY

A simplified OpenFlow flow entry structure:

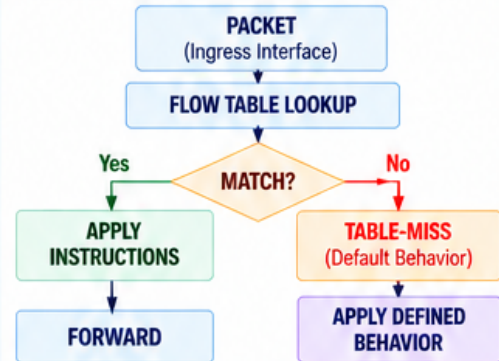
MATCH FIELDS	e.g., src/dst IP, port, VLAN, protocol, etc.
PRIORITY	Defines matching order
INSTRUCTIONS / ACTIONS	Forward, drop, modify, redirect, etc.
COUNTERS	Packets, bytes, duration

Example Actions:



FLOW-BASED FORWARDING

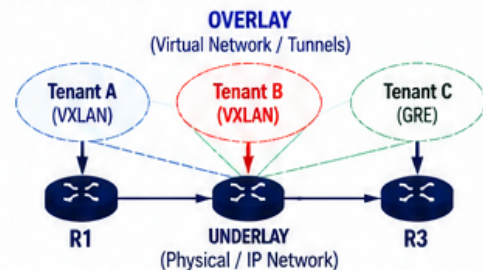
Packets are forwarded based on flow table lookups.



A table-miss entry defines behavior when no higher-priority flow entry matches.

UNDERLAY vs OVERLAY

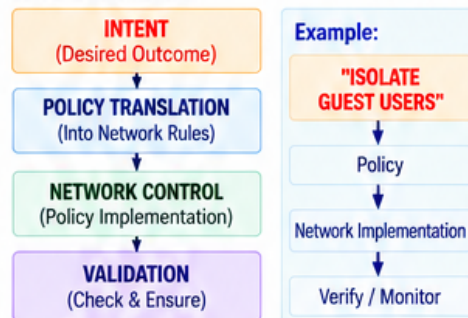
Underlay/overlay networking is common in modern programmable networks, but it is not required for every SDN architecture.



Underlay	Overlay
Physical / IP Network	Logical Network
Provides Reachability	Built Over Underlay
Links, Routers, Switches	Tunnels / Virtual Segments

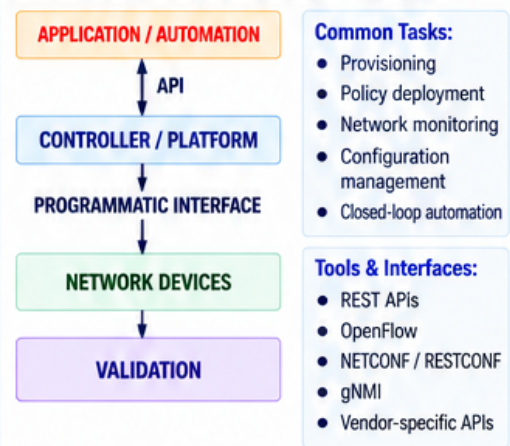
INTENT-BASED NETWORKING

Intent-based networking uses high-level intent to drive network policy and configuration, often leveraging SDN and automation.



Intent-Based Networking and SDN are related concepts, but they are not the same thing.

SDN & NETWORK AUTOMATION



Common Tasks:

- Provisioning
- Policy deployment
- Network monitoring
- Configuration management
- Closed-loop automation

Tools & Interfaces:

- REST APIs
- OpenFlow
- NETCONF / RESTCONF
- gNMI
- Vendor-specific APIs

TRADITIONAL NETWORKING vs SDN

Aspect	Traditional Networking	SDN
Device Approach	Device-Centric	Software / Policy-Driven
Configuration	CLI Common	APIs Common
Control Plane	Distributed Across Devices	Logically Centralized Control
Operations	Device-Level Operations	Network-Wide Abstraction
Changes	Manual Changes Common	Automation-Friendly

SDN COMPLETE FLOW

