Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

AY: 2024-25

Class:	TE	Semester:	V
Course Code:	CSC501	Course Name:	Computer Network

Name of Student:	Sainath Khot
Roll No. :	20
Assignment No.:	6
Title of Assignment:	Software Defined Network
Date of Submission:	
Date of Correction:	

Evaluation

Performance Indicator	Max. Marks	Marks Obtained
Completeness	5	04
Demonstrated Knowledge	3	02
Legibility	2	02
Total	10	08

Performance Indicator	Exceed Expectations (EE)	Meet Expectations (ME)	Below Expectations (BE)
Completeness	5	3-4	1-2
Demonstrated Knowledge Legibility	3	2	1
Legibility	2	1	0

Checked by

Name of Faculty : Miss Sneha Yadav

Signature :

Date : 8/10/24

Assignment 6 (CN)

Q1 Find how fundamental building blocks of software
Defined Networking (SDN) contributing in
separating the central & data planes in network
architecture.

=> This question encourages a broad understanding of
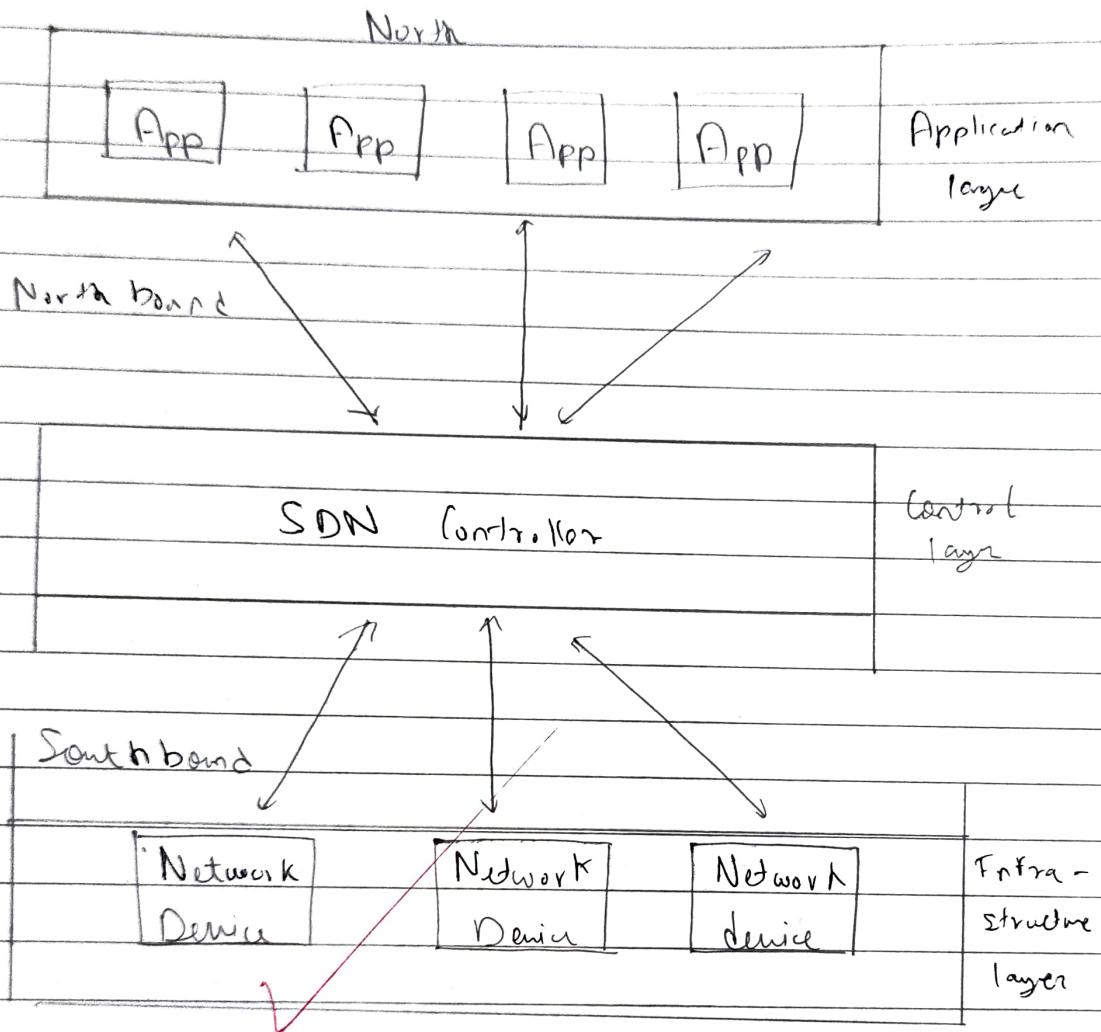SDN architecture (it's key components such as
- Controllers. Centralized counter that manages
the entire network
- Network Devices (Data Plane): Switches & routers
that forward the data based on the instructions
given by the controller.
- Northernbound API's: Interface between the SDN
controller & application, allowing policy & decision.
- South bound API's: Interface between the SDN
controller & application network devices, enabling
the controller to communitate & control the
hardware. It allows the respondent to explain
the functions & relationships between these
components & how they transform traditional
networking & decoupling decision making from
packet forwarding.

FOR EDUCATIONAL USE

Sundaram

SDN Building Block Diagram

North

| App | App | App | App | Application layer |

North bound

SDN Controller — Control layer

South bound

| Network Device | Network Device | Network device | Infrastructure layer |

Q2   Give a network scenario where you need to
implement a software Defined Network SDN with
real time traffic monitoring & custom routing
policies, identify how would you choose
between POX & Nox controllers. Discuss specific
use cases, Advantages & limitations.

=) Nox controller
    Nox is one of the earliest open flow
controllers & is implemented in C++. It is known
for its performance & stability making it suitable
for scenarios where low latency & high throughput
is required.

Advantages:
1) Performance : Since Nox is written in C++, it
offers high performance, making it well
suited for high throughput with minimal
latency requirements. ✓

2) Stability : Nox has been around for a long
time & is considered stable for enterprise grade
implementation

3) Suitable for large scale purposes. Nox can
handle large scale SDN environments due to its
low level access & efficiency.

2. Less flexibility: While NOX is performant, it lacks the flexibility & ease of customisation.

Use case -
1) Large scale Enterprise Network: NOX is a good device if performance & stability are main concerns.

2) Low latency application: NOX is particularly well suited for application that need low latency routing.

2) POX controller
POX is a python based openflow controller which makes it easier to develop & customize compared to NOX

Advantages
1) Ease of Development :- Since POX is written in Python it allows for quicker & simpler development making it easier to implement.
2) Flexibility : POX's modularity & flexibility make it ideal for quick protyping.
3) Active community & has libraries : Consists of available module for traffic enginuring monitoring.
Limitation:
1) Performance :- Since POX is written in py. it is generally slower
2) Not suitable for large scale Environment : Performance limitation macters it slow suitable for large scale.