NYIT

**Fall 2021**

Homework No: 13

Title: Java NOS

**Name:** Patade, Yash Arun

**Class ID#:**  11

**School ID#:** 1284979

**Course:** Java Networking

**Course ID:** CSCI 725

**Date:** 12/14/2021

Assignment Contents

[1) TP Network Router. 3](#_Toc90769472)

[2) Per-router control plane. 3](#_Toc90769473)

[3) SDN Logically Centralized Control Plane. 4](#_Toc90769474)

[4) Software Defined Networking. 4](#_Toc90769475)

[5) Traffic Engineering NOS Application. 5](#_Toc90769476)

[6) Traditional Networking. 5](#_Toc90769477)

[7) What is the OpenDaylight Project? 6](#_Toc90769478)

[8) What is the one word that best describes abstraction? 6](#_Toc90769479)

[9) What is the common mechanism of Java abstraction? 6](#_Toc90769480)

[10) How does abstraction layers help SDN controller development? 7](#_Toc90769481)

[References. 8](#_Toc90769482)

# 1) TP Network Router.

Diagram

Description automatically generated

• Router can be partitioned into control and data plane.

– Management plane used for device configuration.

– Control plane performs decision rules updates (e.g., OSPF Open Shortest Path First).

– Data plane / Forwarding.

# 2) Per-router control plane.

• Individual routing algorithm components in each and every router interact with each other in control plane to compute forwarding tables.

Diagram

Description automatically generated

# 3) SDN Logically Centralized Control Plane.

• A distinct (typically remote) controller server interacts with local Control Agents (CAs) in routers to compute forwarding tables.

A picture containing text, indoor

Description automatically generated

# 4) Software Defined Networking.

• Logically centralized control plane enables:

• Easier network management: avoiding router miss-configurations, greater flexibility of traffic flows.

• Table-based forwarding allowing “programmable” routers.

– Centralized “programming” easier: compute tables centrally and distribute flow tables.

– Distributed “programming” more difficult: compute tables as result of distributed algorithm (protocol) implemented in each and every router.

• Open (non-proprietary and novel) implementation of control plane routing protocols.

# 5) Traffic Engineering NOS Application.

A diagram of a solar system

Description automatically generated with low confidence

• If network operator wants to split heavy u-to-z traffic along parallel routs uvwz and uxyz and perform load balancing, SDN controller can simply instruct u-switch to split traffic (alternate forwarding of arriving frames).

• If network operator wants to divert blue traffic from wz line and allocate wz line to red traffic only, (perform traffic load engineering), SDN controller can simply instruct w switch to use slower and free wy line for blue traffic.

# 6) Traditional Networking.

* Legacy traffic processors (TPs) are smart, loaded with software/firmware and configurable.

– It is very hard to change TP’s software.

Diagram

Description automatically generated

# 7) What is the OpenDaylight Project?

• ODL is a collaborative open source project that aims to accelerate adoption of SDN and create a solid foundation for Network Functions Virtualization (NFV) that offers easy innovation and reduced risk.

• Open SDN framework consisting of code and blueprints.

Diagram

Description automatically generated

# 8) What is the one word that best describes abstraction?

Answer) any.

# 9) What is the common mechanism of Java abstraction?

• Use of public interface that encapsulates whatever internal or lower implementation that may be changed as long as the contractual interface remains unchanged.

– Example: API’s and protocols are interface into the lower level implementation of the used facility.

– Example: Implementation independent model (MDA/MDD MDL and design) abstracts implementation/final-program.

# 10) How does abstraction layers help SDN controller development?

Answer)

• Increased interoperability across various network elements

• Allow greater Innovation, implementation changes & choice

• Ability to select a single solution or best of breed

• Reduced problems of vendor lock-in

# References.

[1] [Index of /02\_725/13\_Java\_NOS (tfbor.com)](http://tfbor.com/02_725/13_Java_NOS/)

[2] [Microsoft PowerPoint - 725\_15\_Lecture\_01\_BISEC2017\_SDNsysDev\_v2.pptx (tfbor.com)](http://tfbor.com/02_725/13_Java_NOS/725_13_Lecture_01_BISEC2017_SDNsysDev_v2.pdf)

[3] [Microsoft PowerPoint - 725\_15\_Lecture\_02\_ODL\_v1.pptx (tfbor.com)](http://tfbor.com/02_725/13_Java_NOS/725_13_Lecture_02_ODL_v1.pdf)