

Providing Layer 2 Data center Interconnection using Overlay technology – VxLAN

CSC - 591 Software Defined Networking

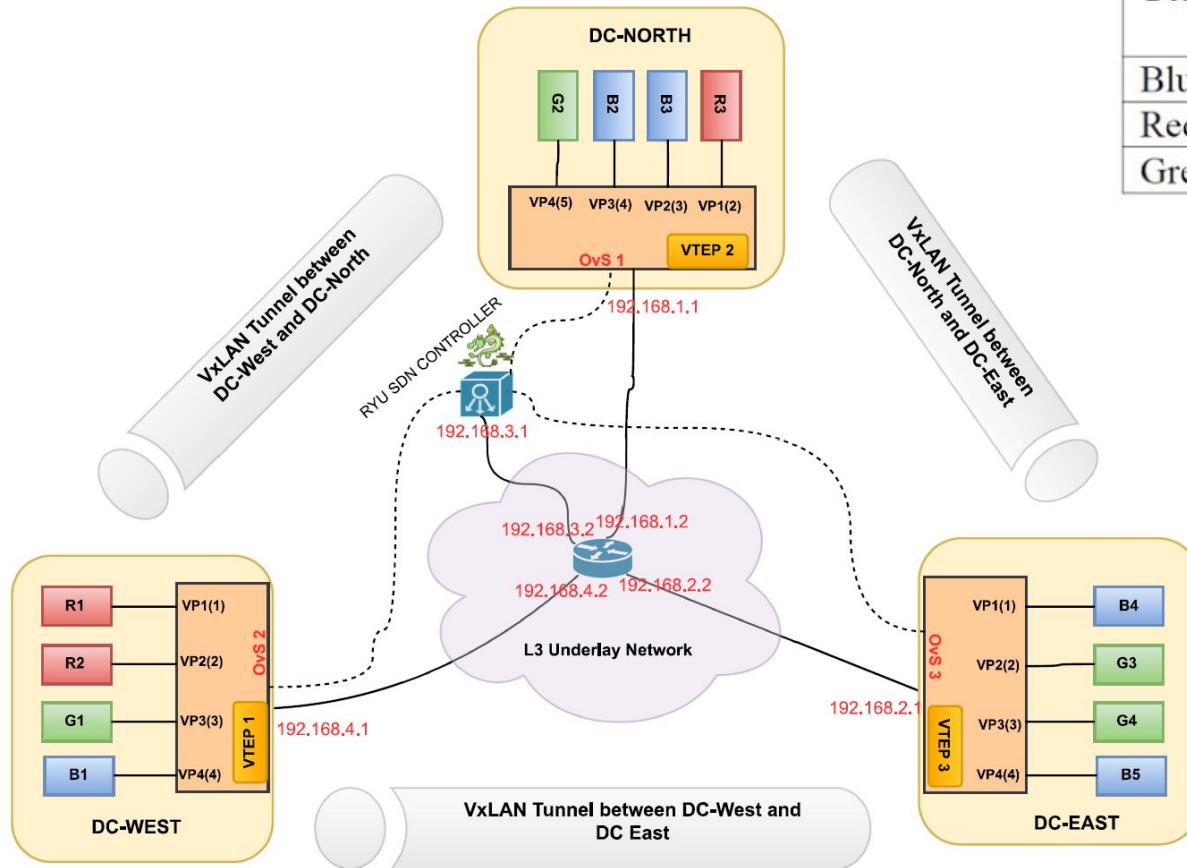
* Ananthram Eklaapuram Lakshmanasamy

* Shishir Bharadwaj Nagendra

* Madhav Bhatt

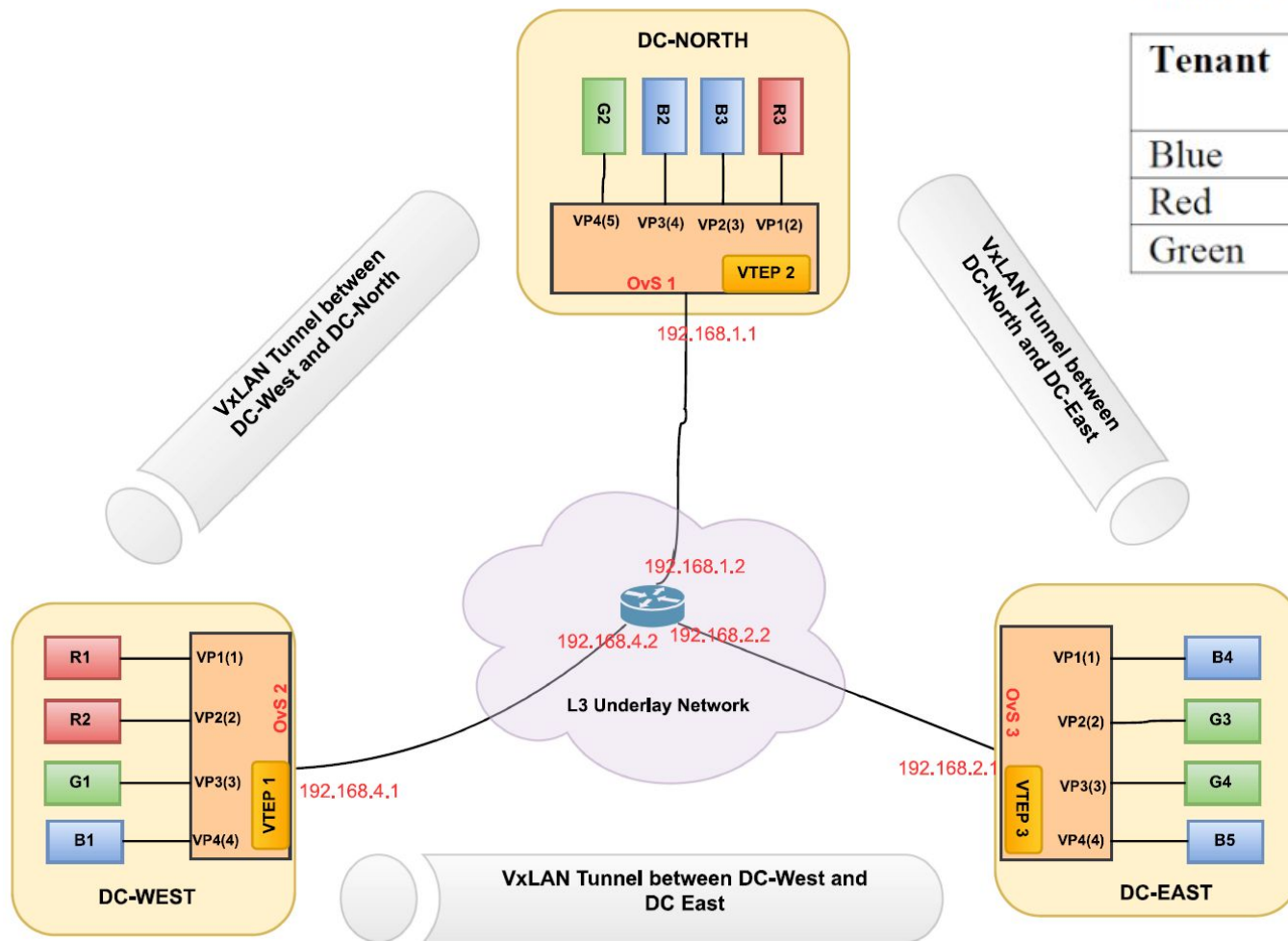
Tenant to VNI mapping

Tenant	VNI
Blue	101
Red	102
Green	103



Tenant to VNI mapping

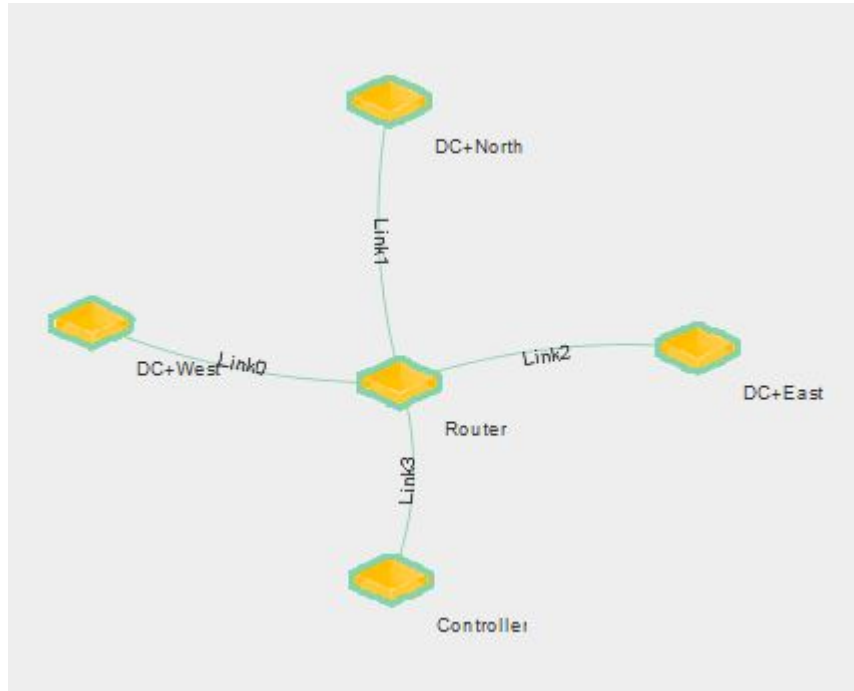
Tenant	VNI
Blue	101
Red	102
Green	103



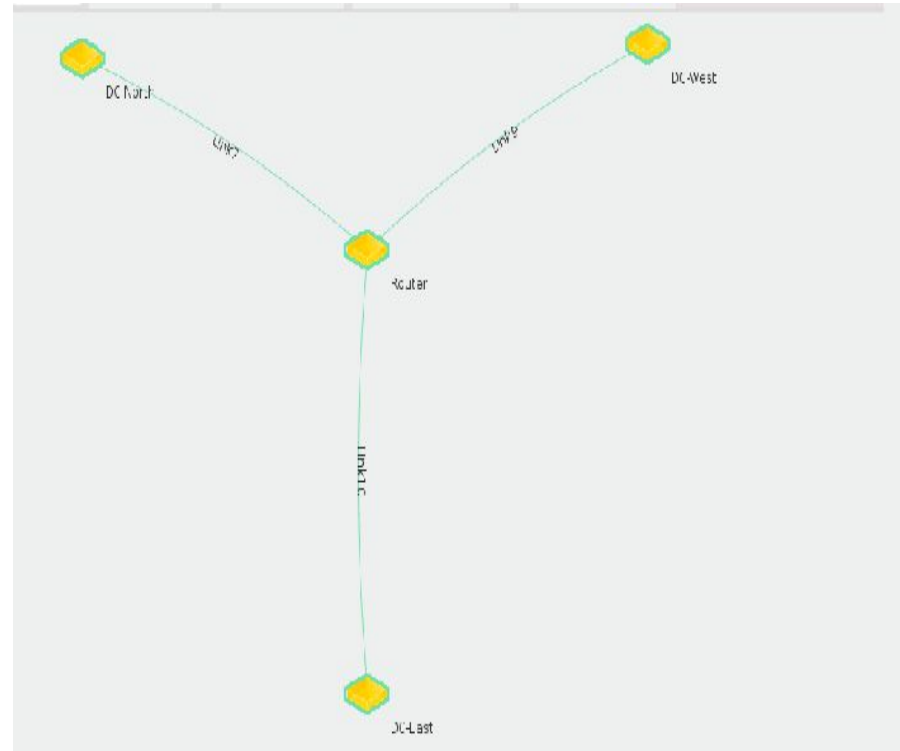
IP Addressing Scheme

Datacenter	VTEP	Tenants	IP Address
DC - West	VTEP 1	R1	10.0.1.1
		R2	10.0.1.2
		G1	10.0.2.1
		B1	10.0.1.1
DC - North	VTEP 2	R3	10.0.1.3
		B3	10.0.1.3
		B2	10.0.1.2
		G2	10.0.2.2
DC - East	VTEP 3	B4	10.0.1.4
		G3	10.0.2.3
		G4	10.0.2.4
		B5	10.0.1.5

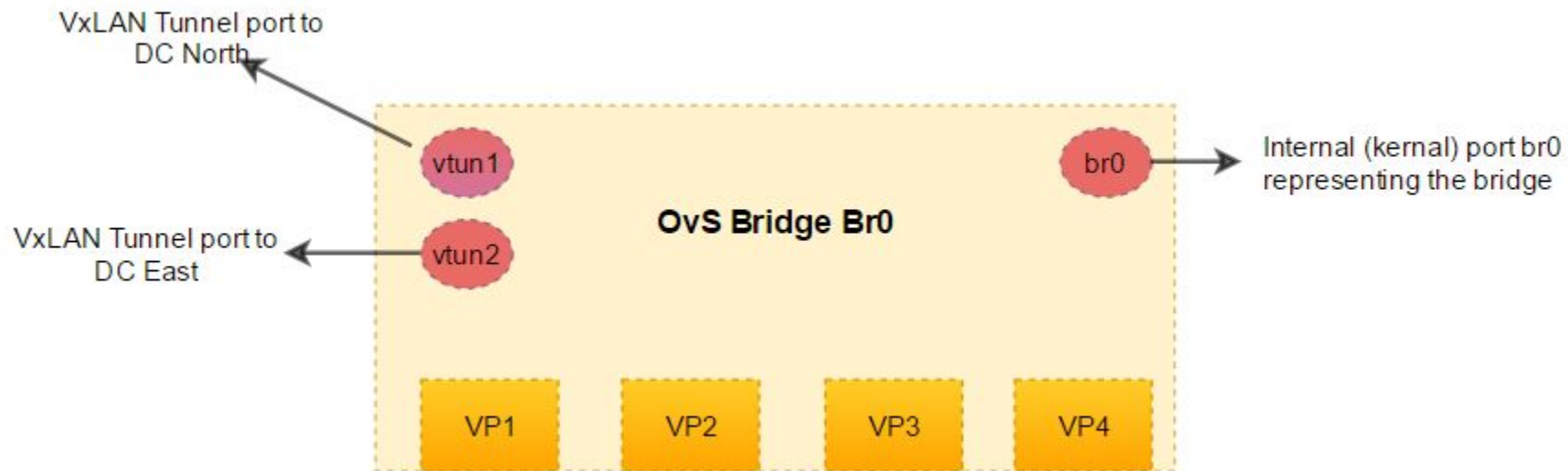
SDN Approach



Traditional Approach



OvS Configuration Logical view



DEMO

D0 (for SDN approach)	Check the Flow Tables at all the OVS Bridges by entering "ovs-ofctl show br0" .	The flow tables are empty	No flows are installed initially and all subsequent additions are going to happen because of the steps we are going to carry out.
--------------------------	---	---------------------------	---

D1	<p>Ping B2 from B1.</p> <p>Capture packets at port eth0 of DC-WEST and vp4 of DC-WEST.</p> <p>Observer the icmp packets at both places.</p>	<p>Packet at eth0 of DC-West is VxLAN encapsulated and packet at vp4 of DC-West is not VxLAN encapsulated</p>	<p>VxLAN encapsulations are working as desired.</p>
----	---	---	---

D2	Set up a HTTP web server at R2(IP: 10.0.1.2) and B2 (IP: 10.0.1.2) which would produce HTTP webpage saying “I am Host R2 and I belong to Tenant Red” and “I am Host B2 and I belong to Tenant Blue” respectively when accessed. Access the webpage (http://10.0.1.2:8080) from R1 and B1.	Different Webpages are loaded at R1 and B1.	Although tenant Blue and tenant Red are having the same IP subnet address space, they are isolated as two different VxLAN broadcast domains. Tenant isolation and address reuse is achieved.
----	---	---	--

<p>D3</p>	<p>VM Addition: Add a new Host VM (say R4 with IP: 10.0.1.4 at DC-East) for an existing tenant at DC.</p> <p>Ping this new VM from other existing VM (R1 at DC-West).</p> <p>VM Deletion: Delete a VM of a tenant (B2 at DC-North with IP 10.0.1.2). Count the number of admin tasks involved in both traditional approach and SDN approach.</p> <p>Try to ping the deleted host (from B1).</p>	<p>Addition and Deletion of VMs is demonstrated</p> <p>Number of steps is less in SDN approach.</p>	<p>SDN solution is more agile.</p>
------------------	---	---	------------------------------------

D4	<p>Tenant Addition: Add a new tenant in all the Datacenters in the Traditional and SDN approach.</p> <p>New Tenant added: Yellow with hosts Y1 (IP: 10.0.3.1) at DC-West and Y2 (IP: 10.0.3.2) at DC-East.</p> <p>Update flows in configs in the both approaches. Ping Y2 from Y1.</p> <p>Tenant Deletion: Remove a tenant from all datacenters in both approaches. Count the Number of admin tasks involved in both cases.</p> <p>Delete the previously added tenant Yellow by updating the configs in both approaches.</p>	<p>For both addition and deletion of a tenant the number of steps executed in the SDN approach will be lesser.</p>	<p>SDN solution is more agile.</p>
----	--	--	------------------------------------

Primary References

1. <https://www.youtube.com/watch?v=tnSkHhsLqpM>

2. <https://inside-openflow.com/2016/07/21/ryu-api-dissecting-simple-switch/>

The image shows a screenshot of a YouTube video player and a blog post. The video player is for a video titled "VXLAN based overlay with Open vSwitch" by David Mahler. The video has 39,031 views and is 10:31 minutes long. The video content includes a list of topics: Configuration of VXLAN tunnel ports in Open vSwitch (OVS), Configuration of OpenFlow entries (OF) in OVS, and Show connectivity for 2 tenants and logical separation of traffic between tenants and from the physical underlay network. Below the video player is a blog post titled "Understanding the Ryu API: Dissecting Simple Switch" by Leo Scott, dated July 21, 2016. The blog post features a diagram titled "Dissecting Simple Switch" showing the interaction between the Ryu OpenFlow API and Python Controller Application Code. The diagram consists of a central box with two arrows pointing in opposite directions, labeled "Ryu OpenFlow API" on the left and "Python Controller Application Code" on the right.

YouTube Red david mahler vxlan

VXLAN based overlay with Open vSwitch

- Configuration of VXLAN tunnel ports in Open vSwitch (OVS)
- Configuration of OpenFlow entries (OF) in OVS
- Show connectivity for 2 tenants and logical separation of traffic between tenants and from the physical underlay network

VXLAN overlay networks with Open vSwitch
David Mahler
Subscribed 14,950 views

Inside OpenFlow
HOME DOWNLOADS

Understanding the Ryu API: Dissecting Simple Switch

by Leo Scott | Jul 21, 2016 | General, OF Core Track, Reference | 0 comments

Dissecting Simple Switch

Ryu OpenFlow API Python Controller Application Code

You are OpenFlow Track
Click this Outline
Ryu for F A quick ov
Creating Workspa Detailed V
Interacth with Post
Custom I Topologi and Introd