

# **Nova/Quantum vif Plugging Interaction**

# Overview

- **Current implementation**
  - Drivers
  - Flow
  - Shortcomings
- **Goals**
  - Clean and robust separation
- **Proposed Solution**
  - Configuration
  - Flow
- **Discussion**

# vif Drivers

- **/etc/nova/nova.conf**

`libvirt_vif_driver=nova.virt.libvirt.vif.LibvirtHybridOVSBridgeDriver`

- **Driver**

*class **VIFDriver**(object):*

*def **plug**(self, instance, vif, \*\*kwargs):*

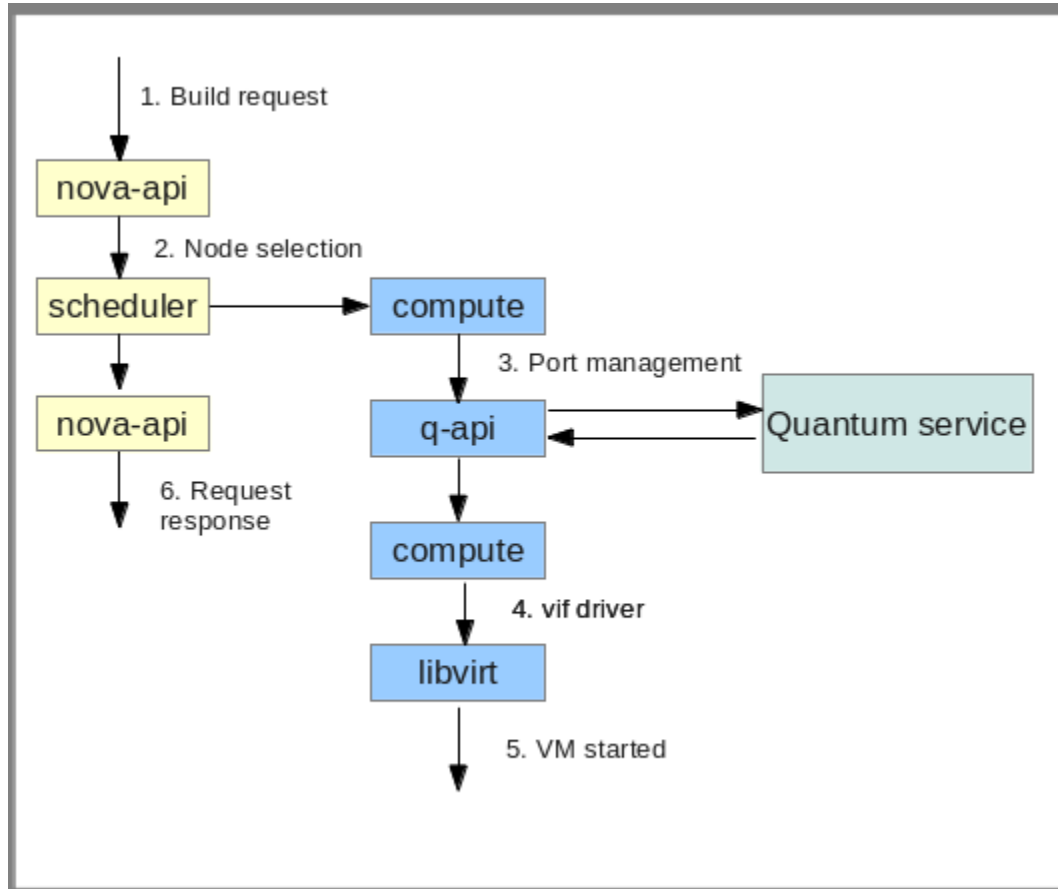
*"""Plug VIF into network."""*

*def **unplug**(self, instance, vif, \*\*kwargs):*

*"""Unplug VIF from network."""*

- specific per (plugin, virt driver) pair
- returns network configuration
- may configure networking objects

# Flow



# Shortcomings

- Interface type 'ethernet' lowers level of host protection
  - Unable to optimize tap device without bridge (LinuxBridge)
- New plugins may require new vif driver
- VIF configuration may vary per hypervisor type
- Error handling and fault tolerance
  - Example: agent is unable to create bridge
- Quantum + Nova are mutually dependant requiring synchronized changes
  - Unit testing across applications

# Proposal

- Separation between configuration and networking
- Configuration
  - Quantum *port* create returns configuration details
    - done via extensions
    - may differ per network type
  - External controller updates via Quantum plugin
  - No need for *libvirt\_vif\_driver* per plugin
- Networking
  - performed by Quantum agent/library (if there is no agent)

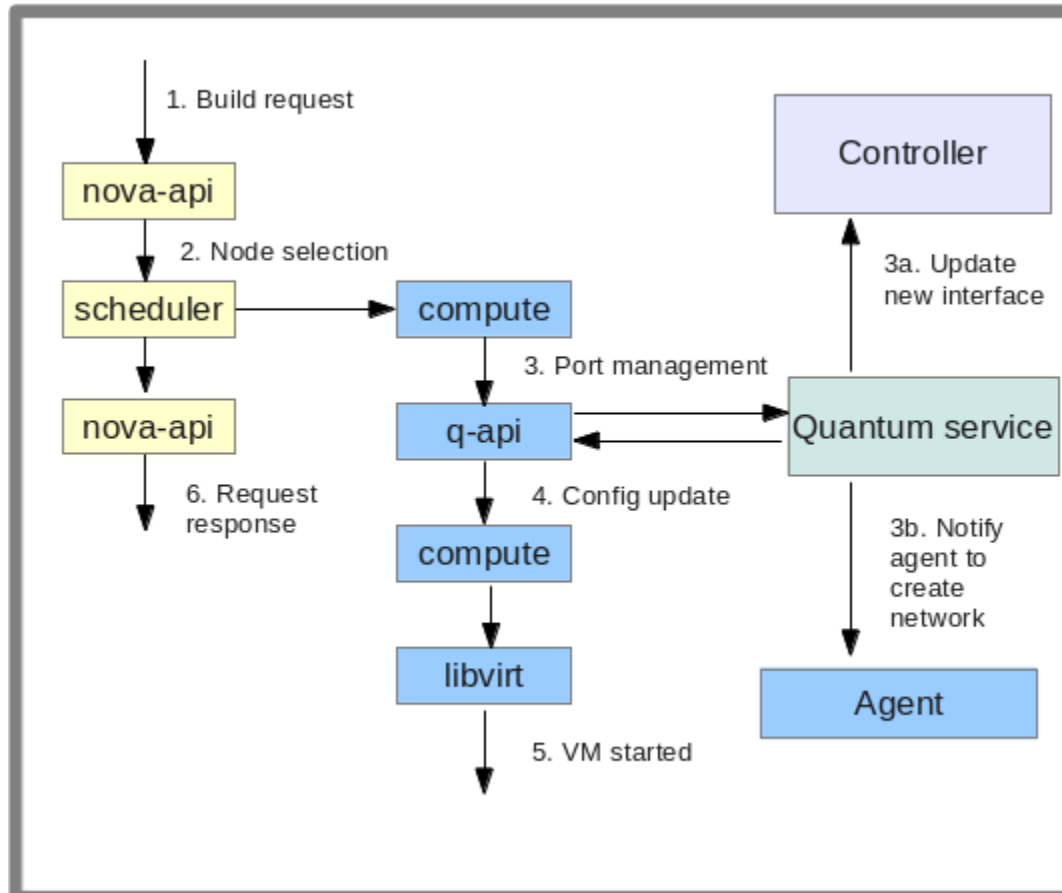
# Configuration Options

- Only 4 possible vif configurations

Type	Required parameters
Linux Bridge	bridge name
OVS Bridge	bridge name and port profiles
macvtap 802.11Qbg	eth device name and port profile parameters
macvtap 802.11Qbh	eth device name and port profile parameters

*Note: libvirt will honour tap device names*

# Flow





# Discussion

- Other options
  - Fixing/tweaking current code
- Integration issues