

Cloud Orchestration Networking

Increasing server-to-server speed, and increased scalability

The diagram illustrates the ciao Networking architecture, showing the flow of traffic and control signals between different components:

- Network Node(s):** The top left component, which connects to the Network Node (NN) via "Tunnels to Upstream CNs".
- Network Node (NN):** The top center component, which connects to the Control Nodes via "SSNTP".
- Control Nodes:** The top right component, which contains the "ciao-controller" and "ciao-scheduler".
- Local CNCI - Tenant 1 and Local CNCI - Tenant N:** These are the central components for each tenant, containing "CNCI Agent" and "Subnet 1 bridge" / "Subnet N bridge".
- Compute Node (CN):** The bottom components, each containing a "ciao-launcher" and one or more "Ins" (Instances) connected to "Subnet 1" and "Subnet N".

The diagram shows the following connections:

- Control Plane (SSNTP):** Solid orange arrows from the Network Node (NN) to the ciao-scheduler, and from the ciao-scheduler to the ciao-launchers in the Compute Nodes.
- Data Plane (Tunnels to CNCI):** Dashed blue arrows from the ciao-launchers in the Compute Nodes to the Local CNCI - Tenant 1 and Local CNCI - Tenant N components.
- Local Connections:** Solid orange arrows from the Local CNCI - Tenant 1 and Local CNCI - Tenant N components to the ciao-launchers in the Compute Nodes.

Findings and Results

Technical Details

- Simpler Software Defined Network (SDN) implementation. An SDN is a network layer defined in software rather than on physical switches and routers.
- Targets small or medium enterprise datacenters with several hundred servers
- Creates self-configuring, stateless, software defined overlay networks for tenants on top of existing networking hardware
- Uses Linux bridges and GRE tunnels to create overlays
- Problems:
 - Current SDN innovation is being done with Open vSwitch
 - Linux bridges and GRE tunnels used by Ciao are not compatible with Open vSwitch

The diagram illustrates the network architecture for Host 1 and Host 2. Host 1 contains VM1 and VM2, which connect to tap0 and tap1 respectively. These taps connect to an OVS bridge, which is connected to eth0 and eth1. Host 2 contains VM3 and VM4, which connect to tap0 and tap1 respectively. These taps connect to an OVS bridge, which is connected to eth0 and eth1. The eth0 interfaces of both hosts are connected to the Transport Network, and the eth1 interfaces are connected to the Management Network.

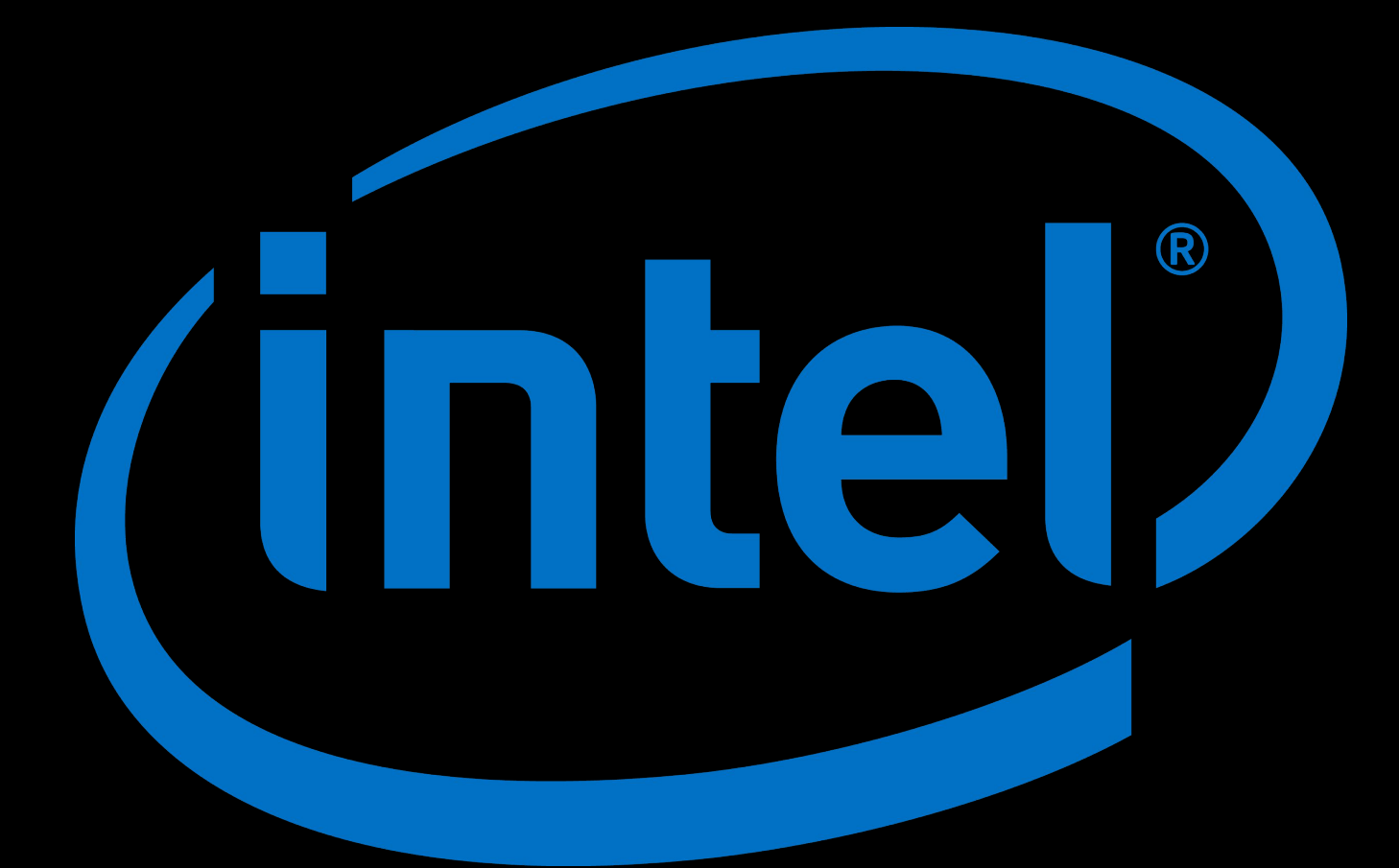
Source: http://docs.openvswitch.org/en/latest/_images/tunneling.png

Project Members and Sponsor

Garrett Smith, Computer Science
smithgar@oregonstate.edu

Cody Malick, Computer Science
malickc@oregonstate.edu

Matthew Johnson, Computer Science
johnsma8@oregonstate.edu



Robert Nesius, Engineering Manager, OTC
Advanced Systems Engineering, Intel
Corporation

Special thanks to Intel Principal Engineer
Manohar Castelino