Andromeda
Performance, Isolation, and Velocity at Scale in Cloud Network Virtualization
Andromeda Goals

Performance and Isolation
High throughput and low latency, regardless of the actions of other tenants

Velocity
Quickly develop and deploy new features and performance improvements

Scalability
Large networks, many tenants, rapid provisioning
Andromeda Architecture

VM Controller

OpenFlow Front End

Extended OpenFlow

VM Host

Virtual switch

Match | Actions
--- | ---
Green | Red
Green | Green
Green | Green
Red | Red
Red | Red
Red | Red

RPC
Andromeda Architecture

VM Controller -> OpenFlow Front End

Extended OpenFlow

OpenFlow Front End -> VM Controller

RPC

VM Host

VM

VM

VM

VM

VM

Virtual switch

Match | Actions

1. New VM added
2. Install flows from other VMs to the new VM.
1. New VM added

2. Install flows from other VMs to the new VM.

3. Install flows from the new VM to other VMs in the network.
Scaling Goals

Global connectivity

Large virtual networks (100k+ VMs)

Rapid provisioning
  Enable on-demand workloads
Programming Time for Large Networks

Setup:
- VMs are placed on 10,000 hosts
- 30 VM Controller partitions

Programming time is $O(n \times H)$
- $n =$ number of VMs
- $H =$ number of hosts

Quadratic scaling leads to provisioning challenges
- Control plane CPU and memory
- Dataplane memory
Scaling with Hoverboards

Host 10.1.1.3  
vmA  
vmX  
vmY

virtual switch

low priority route
Hoverboard

Host 10.1.2.5  
virtual switch  
vmE  
vmZ  
vmV

Host 10.1.2.4  
virtual switch  
vmB  
vmC  
vmD
Hoverboard Offloading

- Host 10.1.1.3
  - vmA
  - vmX
  - vmY
  - virtual switch

- Host 10.1.2.5
  - virtual switch
  - vmE
  - vmZ
  - vmV

- Host 10.1.2.4
  - virtual switch
  - vmB
  - vmC
  - vmD

vmX → vmZ offload flow

low priority route
Hoverboard Offloading

VM Controller

OpenFlow Front End

flow programming

vmX → vmZ offload flow

low priority route

Hoverboard

Host 10.1.1.3

vmA

vmX

vmY

virtual switch

stats

Host 10.1.2.5

virtual switch

vmE

vmZ

vmV

Host 10.1.2.4

virtual switch

vmB

vmC

vmD

low priority route

vmX → vmZ offload flow

flow programming

stats
Programming Time for Large Networks

Hoverboards reduce time to program network connectivity for large networks
- 37X faster for a 40,000-VM network
Why Hoverboards Are Effective

**Peak throughput** for all VM pairs in all virtual networks in one cluster over a 30-minute interval

Today, **more than 99.5% of traffic is offloaded.**
Andromeda Data Plane

OS bypass, busy polling dedicated CPU Fast Path for **high performance**

Userspace dataplane, live migration, and hitless upgrades for **feature velocity**
Andromeda Data Plane

OS bypass, busy polling dedicated CPU Fast Path for **high performance**

Userspace dataplane, live migration, and hitless upgrades for **feature velocity**
Andromeda Data Plane

OS bypass, busy polling dedicated CPU Fast Path for **high performance**

Userspace dataplane, live migration, and hitless upgrades for **feature velocity**

Busy polls physical & virtual NIC queues, forwards VM packets
Andromeda Data Plane

OS bypass, busy polling dedicated CPU Fast Path for **high performance**

Userspace dataplane, live migration, and hitless upgrades for **feature velocity**

- **OpenFlow Front End**
- **Extended OpenFlow**
- **Open vSwitch**
- **Management Plane**
- **Andromeda Fast Path**
  - Match
  - Action
  - Flow cache
- **Host OS Kernel**

**Routes packet, applies per-flow Fast Path actions (encap, decap, etc)**
Andromeda Data Plane

OS bypass, busy polling dedicated CPU Fast Path for high performance

Userspace dataplane, live migration, and hitless upgrades for feature velocity
Andromeda Data Plane

Fast Path polls guest VM rings & copies packets to/from guest VM memory

OS bypass, busy polling dedicated CPU Fast Path for **high performance**

Userspace dataplane, live migration, and hitless upgrades for **feature velocity**
Data Plane - Fast Path

High performance traffic processed end-to-end on **Fast Path**

- > 30Gb/s throughput & > 3M pps on one core
- Flow Table performs routing, encap/decap, etc.
- Fast Path polls virtual & physical NIC rings
Data Plane - Fast Path

Extended OpenFlow

OpenFlow Front End

High performance traffic processed end-to-end on Fast Path

> 30Gb/s throughput &
> 3M pps on one core

Flow Table performs routing, encap/decap, etc.

Fast Path polls virtual & physical NIC rings

Andromeda Fast Path

Flow Lookup
Route, decap, ...

Match Action

Packet

VM Host

Guest VM

Guest VM Coprocessor

Open vSwitch

Management Plane

Host OS Kernel

Google Cloud
Data Plane - Fast Path

High performance traffic processed end-to-end on Fast Path

> 30Gb/s throughput &
> 3M pps on one core

Flow Table performs
routing, encap/decap,
etc.

Fast Path polls virtual &
physical NIC rings
Data Plane - Coprocessor Path

- **Guest VM**
- **VM Host**
- **Guest VM Coprocessor**
- **Open vSwitch**
- **Management Plane**
- **OpenFlow Front End**
- **Extended OpenFlow**
- **Andromeda Fast Path**
- **Host OS Kernel**

**Coprocessors** are per-VM threads CPU attributed to VM container.

Coprocessors execute CPU-intensive packet ops such as DoS.

Decouples feature growth from Fast Path speed.
Data Plane - Coprocessor Path

Coprocessors are per-VM threads CPU attributed to VM container

Coprocessors execute CPU-intensive packet ops such as DoS

Decouples feature growth from Fast Path speed
**Data Plane - Coprocessor Path**

**Coprocessors** are per-VM threads CPU attributed to VM container

Coprocessors execute CPU-intensive packet ops such as DoS

Decouples feature growth from Fast Path speed
Data Plane - Coprocessor Path

Coprocessors are per-VM threads CPU attributed to VM container

Coprocessors execute CPU-intensive packet ops such as DoS

Decouples feature growth from Fast Path speed

Deliver packet to VM
Copy, update rings, ...

OpenFlow Front End
Extended OpenFlow

VM Host
Guest VM
Packet
Guest VM Coprocessor

Andromeda Fast Path
Host OS Kernel

Open vSwitch
Management Plane

Miss
Insert

Match
Action
Flow
Cache

NIC

Flow cache

Host OS Kernel

Google Cloud
VM-VM Throughput

**Single core** per host for dataplane Fast Path. Skylake testbed hosts.

Both hosts connected to same Top of Rack switch.
VM-VM Round Trip Latency

Single core per host for dataplane Fast Path. Skylake testbed hosts.

Both hosts connected to same Top of Rack switch.
CPU Efficiency

Minimizing host (and guest) network CPU cycles per byte (CPB) is critical.

Since initial production release, we have improved CPB by $>16x$ as measured on sender + receiver host during a multi-stream benchmark.

Andromeda 2.0+ use a single core per host for the dataplane Fast Path. Results from Sandybridge testbed hosts connected to same ToR switch.
CPU Efficiency Evolution

- **Andromeda 1.0**: Kernel datapath
- **Andromeda 1.5**: Optimize pipeline
- **Andromeda 2.0**: OS bypass, 1 thread hop
- **Andromeda 2.1**: Remove thread hop
- **Andromeda 2.2**: Memory copy offload

<table>
<thead>
<tr>
<th>Version</th>
<th>Host</th>
<th>Guest</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre Andromeda</td>
<td>43.5</td>
<td>16.0</td>
</tr>
<tr>
<td>Andromeda 1.0</td>
<td>30.4</td>
<td>12.3</td>
</tr>
<tr>
<td>Andromeda 1.5</td>
<td>5.4</td>
<td>5.6</td>
</tr>
<tr>
<td>Andromeda 2.0</td>
<td>2.6</td>
<td>5.0</td>
</tr>
<tr>
<td>Andromeda 2.1</td>
<td>2.0</td>
<td>4.9</td>
</tr>
<tr>
<td>Andromeda 2.2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Velocity

A rapid release cycle enables swift deployment of features & bug fixes.

Our dataplane has weekly rollouts via non-disruptive upgrades.

Live migration allows VMs to be migrated between physical host without disruption, enabling transparent host maintenance.
Dataplane Hitless Upgrade 1 / 3

Upgrade Brownout

Old Dataplane state is transferred to New Dataplane in the background

Old Dataplane continues serving physical NIC & virtual NIC queues

Guest VM

Old Dataplane

State Xfer

New Dataplane

Physical NIC
Dataplane Hitless Upgrade 2 / 3

Upgrade Blackout

Old Dataplane stops serving virtual & physical NIC queues

Then, any updated (delta) Old Dataplane state is transferred to New Dataplane

Guest VM

Old Dataplane

State Xfer

New Dataplane

Physical NIC
Dataplane Hitless Upgrade 3 / 3

Upgrade Complete

State xfer done. Median blackout time is \textbf{270ms}.

New Dataplane starts serving VM virtual NIC & physical NIC queues

Old dataplane terminated
Conclusion

We have discussed the design and evolution of Andromeda

Control plane scalability & Rapid provisioning

• Hoverboard model avoids programming long tail of mostly idle flows on VM host. Scales to 100k VMs/network

High performance & Feature velocity

• OS Bypass dedicated CPU dataplane provides high performance (> 30Gb/s, > 3M pps with 1 core) & weekly non-disruptive updates