Execution Environments for Distributed Computing

NaaS: Networking as a Service, a new business model for network operators

Master in Computer Architecture, Networks and Systems - CANS

EEDC Project
Group number: EEDC-2
Group members:
Muhammad Anis uddin Nasir
Emmanouil Dimogerontakis
Outline

● NaaS
● NaaS & Cloud
● Solution Approaches
● The Future of NaaS
Outline

● NaaS
● NaaS & Cloud
● Solution Approaches
● The Future of NaaS
The Service Trend

Everything as a Service

Humans as a Service

Software as a Service

Platform as a Service

Infrastructure as a Service

Hardware Layer
"Decoupling infrastructure management from service management can lead to innovation, new business models, and a reduction in the complexity of running services. It is happening in the world of computing, and is poised to happen in networking."

Jennifer Rexford, Eric Keller
Outline

● NaaS
● **NaaS & Cloud**
  ○ Cloud Limitations
  ○ Customer Interests
  ○ Other Possibilities
● NaaS Solutions
● **The Future of NaaS**
Cloud Limitations

NaaS & Cloud

SPEED LIMIT 15
Limited visibility of network resources

- No network monitoring
- No multicast services
- No custom path selection

Limited Security

- No VPN establishment
- Limited Network policies
- Lack middlebox interposition
Customer Interests  NaaS & Cloud

- Configure specific network policies
- Access to Data Plane
- Implement in-network services
- Performance
Distinct network services can be provided by

- Cloud-providers
- Third-party providers
- Research approaches
- Network Virtualization (Overlay Networks etc.)

But a central efficient framework does not exist.
Outline

● NaaS
● NaaS & Cloud
● NaaS Solutions
  ○ The Single Router
  ○ NaaS Boxes
  ○ CloudNaaS
● The Future of NaaS
"Basic Connectivity is Not Enough"

**Goal**: Provide in-network functionality for
- Customer controlled routing
- Cloud computing
- Gaming and Live video streaming
- Network Monitoring
The Single Router  NaaS Solutions

What it provides?

- Interactive Program (API)
- Routing Policies
- Access to Data Plane
- General-purpose processing
The Single Router NaaS Solutions

Customer Program

Routing Software

General purpose functions

Data Plane

The single router abstraction.
The Single Router  NaaS Solutions

Challenge:

Infrastructure is distributed

Solutions:

- Choose a placement
- Configure inter-processor communication
- "Tune" the switch
- Dynamic Adaptation
The Single Router  NaaS Solutions

Challenge:
Infrastructure is shared

Solutions:
- Single control process handling all customer's routing sessions
- Tag with customer ID and process each using the particular customers policy
Benefits

- Automate configuration
- Manage separate services independently
- Simplify management
- Outsourced IT
Goal: Enable cloud users to

- deploy custom routing protocols
- deploy custom multicast protocols
- modify packets' content on path
<table>
<thead>
<tr>
<th>Application</th>
<th>Current approach</th>
<th>Mem</th>
<th>Per-packet state</th>
<th>Packet operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Broadcast/multicast [14,33]</td>
<td>overlay</td>
<td>KBs</td>
<td>✗</td>
<td>duplicate</td>
</tr>
<tr>
<td>Content-based netw. [13]</td>
<td>overlay</td>
<td>MBs</td>
<td>✗</td>
<td>duplicate</td>
</tr>
<tr>
<td>Content-centric netw. [22]</td>
<td>overlay</td>
<td>MBs</td>
<td>✗</td>
<td>duplicate</td>
</tr>
<tr>
<td>Complex event processing</td>
<td>overlay</td>
<td>GBs</td>
<td>✓</td>
<td>modify</td>
</tr>
<tr>
<td>Stream processing [15]</td>
<td>overlay</td>
<td>GBs</td>
<td>✓</td>
<td>modify</td>
</tr>
<tr>
<td>Data aggregation [4,35]</td>
<td>overlay</td>
<td>MBs</td>
<td>✓</td>
<td>modify</td>
</tr>
<tr>
<td>Deduplication [8]</td>
<td>end-to-end</td>
<td>GBs</td>
<td>✓</td>
<td>modify</td>
</tr>
<tr>
<td>Distributed caching [3]</td>
<td>end-to-end</td>
<td>GBs</td>
<td>✓</td>
<td>modify</td>
</tr>
<tr>
<td>Information flow control [28]</td>
<td>middlebox</td>
<td>KBs</td>
<td>✗</td>
<td>forward</td>
</tr>
<tr>
<td>Stateful firewalls/IDS</td>
<td>middlebox</td>
<td>MBs</td>
<td>✓</td>
<td>forward</td>
</tr>
<tr>
<td>Packet scheduling [14]</td>
<td>middlebox</td>
<td>MBs</td>
<td>✓</td>
<td>forward</td>
</tr>
<tr>
<td>Multipath [30]</td>
<td>network</td>
<td>KBs</td>
<td>✓</td>
<td>forward</td>
</tr>
<tr>
<td>Load-aware anycast</td>
<td>various</td>
<td>KBs</td>
<td>✓</td>
<td>forward</td>
</tr>
</tbody>
</table>
NaaS Boxes

Functionality
- Network Visibility
- Custom Forwarding
- In-network Processing

Requirements
- Integration with current DC hardware
- High-level Programming Model
- Scalability and isolation
Benefits

- Fine grained control over small fraction of applications:
  - Improve overall network efficiency
  - Improve performance for individual customer
- Considers limited processing capabilities of network components
SDN + Indirection + Host-Based vSwitches

**Goal:** Deploy extensible set of network functions such as

- fine-grained network isolation
- custom addressing
- flexible interposition of various middleboxes
- optimizations for performance and availability
CloudNaaS

NaaS Solutions

(a) Specify user requirements
(b) Convert requirements into a communication matrix
(c) Compile matrix entries into network-level rules
(d) Install rules and configure paths
Cloud Controller

- Specify user requirements
- Convert rules into communication matrix
- Modified

CloudNaaS NaaS Solutions

User network policy
Cloud Controller
(a) Specify user requirements

User policy
Communication matrix
User allocated resources
(b) Convert requirements into a communication matrix

OPEN NEBULA
CloudNaaS NaaS Solutions

**Network Controller**

- Compile matrix entries into network-level rules
- Install rules and configure paths

- [OpenFlow](https://openflow.org)
- [nox](https://noxrepo.readthedocs.io)
- [Open VSwitch](https://openvswitch.org)

(c) Compile matrix entries into network-level rules
(d) Install rules and configure paths
address dbserver1 = {128.1.104.103}
address dbserver2 = {128.1.104.13}
group frontend = {httpdserver}
group businesslogic = {jboss1,jboss2, jboss3}
group backend = {dbserver1, dbserver2}
middlebox Class = {type=classifier, config=""}  
middlebox DPI = {type=dpi, config=""}  
networkservice protectFrontEnd = {12broadcast=no, qos=standard, mb=DPI}  
networkservice connectBL = {12broadcast=no, qos=standard, mb=none}  
networkservice reservedBW = {12broadcast=no, qos=10mbs, mb=Class}  
networkservice allowFailover = {12broadcast=yes, qos=standard, mb=none}  
virtualnet allowFailover (backend)  
virtualnet protectFrontEnd(frontend, EXTERNAL)  
virtualnet connectBL(frontend,businesslogic)  
virtualnet reservedBW(businesslogic,backend)
Benefits

● Allows network functions for production enterprise applications in IaaS clouds
● Optimized for use in cloud
● Supported by experiments and simulations
● Innovative networking technology and standards
● Existing prototype
Outline

● NaaS
● NaaS & Cloud
● NaaS Solutions
● The Future of NaaS
Future of NaaS

Opportunities:

- New business model
- New research horizons regarding NaaS:
  - Scalability, Performance, Isolation, Programmability
  - Pricing model
  - WAN extension
Conclusion

Network as a Service

● Provide to the users access to network infrastructure
● Must have: High level API, Security, Scalability
● Issues: Performance, Isolation, Programmability
● Different approaches proposed - still open research area
● NaaS can lead to new business models
References


Special Thanks To
NaaS: Networking as a Service, a new business model for network operators

EEDC Project
Group number: EEDC-2
Group members:
Muhammad Anis uddin Nasir
Emmanouil Dimogerontakis