An OpenNaaS based SDN Framework for Dynamic QoS control

Jordi Ferrer Riera
Research Engineer

i2cat

SDN for Future Network Services, Trento, Nov. 2013
An OpenNaaS based SDN Framework for Dynamic QoS control

Iris Bueno, José I. Aznar, Eduard Escalona, Joan A. García-Espín, and Jordi Ferrer Riera
1. Introduction
   • Rationale & Motivation
   • State of the art
2. Network Control Layer
   • Architecture and Design
   • Components
   • Implementation over OpenNaaS
3. Use Case
4. Conclusion
INTRODUCTION
Introduction

OTT services

High-consume apps

On-line gaming

QoS provided by ISPs and Network operator

Dynamic and customized QoS provisioning

REQUIRE

SDN for Future Network Services, Trento, Nov. 2013
Introduction

- Network limitations...
  - Lack of strict, and tailored levels of e2e QoS
  - Best-effort network
    - Unpredictability
    - Unreliability
  - No real control of the utilisation of the resources
    - Overprovisioning
  - Lack of flexibility
    - On-demand provisioning of resources
    - Static and non-programmable network

SDN for Future Network Services, Trento, Nov. 2013
What happened...?

- **Best-effort**
  - Unpredictable behaviour
  - Poor quality
- **IntServ**
  - Many stored states: scalability
- **DiffServ**
  - Aggregated traffic
- **DiffServ over MPLS**
- **CAC**
  - VoIP applications
  - Scalability

None is completely extended to provide e2e QoS
None supports dynamic on-demand QoS delivery
What happens today...?

- **Network as a Service (NaaS)**
  - Business model for delivering tailored on-demand *network services* over dedicated infrastructure (virtualized or not).
  - Management model related to network infrastructure servicing based on *resource* and *capabilities*.

- **Resource**
  - *Logical representation of a physical or virtual device*.

- **Capability**
  - *Features associated to a given resource*.

- **Software Defined Networking**
  - Based on *data and control plane decoupling*.
  - *Global knowledge* at the controller, where the decisions are taken.
What do we propose...

**FEATURES**

- Separates (and abstracts) the control and data plane
- External controller takes the network decisions
- Customized TE algorithms
- Virtualization of network resources
- Monitoring of networking resources

**ENABLES**

- Control over particular QoS related parameters
- Allows dynamically reconfiguring the network
- Optimization of the utilisation of network resources
- Adaptive and customized routing of data streams

**Flexibility, reliability and adaptability to the QoS requirements of applications and services**
NETWORK CONTROL LAYER
The Network Intelligence

- **SDNApp**: Set of generic modular functions over an OF network. It adapts control plane to fit service requirements.
- **SDNMonitor**: Monitoring tools to measure the status of the network and retrieve information.
- **SDNController**: Responsible for the configuration of the actual devices.
- **Three interfaces**

SDN for Future Network Services, Trento, Nov. 2013
SDNApp

- Builds an updated topology map, which reflects status of the network resources and the QoS levels of the different running apps
  - Resource allocator
  - Provisioning
  - QoS tracker
  - Application request engine
  - Algorithm TEDB

SDNMonitor

- Retrieve network state information and statistics through OpenFlow counters
- Deliver network statistics to the SDNApp

NCL External Interfaces

- Northbound (service request + monitor statistics)
- Inter-Domain (NCL to NCL)
- Southbound (device config + monitor status)
How do we implement it...?

Network Intelligence
- Integration with Northbound Middleware
  - IaaS/Cloud managers
  - Other NMS.
- The user

NaaS Layer
- Network HAL abstraction to infrastructure.
- Resources manageable by the user.
  - Access controlled by the Sec. Manager.

Platform
- Reusable building blocks, common to all extensions.
- Controls access to the infrastructure.
  - Integrity, Policy, etc..

Managed infrastructure

Visit http://www.opennaas.org

SDN for Future Network Services, Trento, Nov. 2013
How do we implement it...?

Simple extensions...!
Software Architecture

REST API

OpenNaaS Framework

OpenNaaS Plugins

SDNapp

MONITOR

SDN NETWORK RESOURCE

OPENFLOW SWITCH RESOURCE

REST API

OPENFLOW PROXY

External interface

Physical network resources

Middleware

Extensions

OpenFlow

OpenFlow

OpenFlow

SDN for Future Network Services, Trento, Nov. 2013
USE CASE
FP7 OFERTIE

- OpenFlow Experiment in Real-Time Internet Edutainment (OFERTIE) FP7 project
  - OFERTIE aims to use software-defined networking approaches to improve delivery of an emerging class of distributed applications for the Future Internet known as Real-Time Online Interactive Applications (ROIA).
  - OFERTIE aims to enhance and use the OFELIA Test-bed for OpenFlow Programmable Networking to run experiments
  - Duration: 24M
  - Partners: University of Southampton, University of Munster, i2CAT, Spinor, Turk Telekom, Sobee, Interoute
CONCLUSION
Conclusion

- There is a gap between the QoS level required by current applications (e.g. online real-time gaming) and the QoS offered by the ISPs
  - Several QoS schemes proposed

- Network Control Layer
  - SDNApp responsible for QoS
  - SDN Controller
  - SDN Monitor

- OpenNaaS as the key enabler for SDNApp implementation

- Developments on-going within FP7 OFERTIE project
Thanks!
Moltes Gràcies!

jordi.ferrer@i2cat.net
jordi.ferrer.mntc

Questions?