NUBOMEDIA: an Elastic PaaS Enabling the Convergence of Real-Time and Big Data Multimedia

SmartCloud 2016

19th November 2016 (New York, USA)

Boni García

Universidad Rey Juan Carlos (Spain)

boni.garcia@urjc.es

Table of contents

Table of contents

1. Introduction

Problem at hand

- Multimedia applications and services are becoming the main force of the Internet
  - For example: WebRTC, Video Content Analysis (VCA) and Augmented Reality (AR)
- Deploying these types of technologies in common clouds infrastructures is complex and cannot be achieved easily

1. Introduction

Our proposal: NUBOMEDIA

- NUBOMEDIA is an open source **PaaS** (Platform as a Service)
- NUBOMEDIA exposes to developers the ability of **deploying and leveraging** applications with media capabilities:
  - WebRTC, media recording, group communications, VCA, AR, etc.

1. Introduction

Our proposal: NUBOMEDIA

- NUBOMEDIA has been conceived for simplifying the way developers use to deal with multimedia applications
  - From the developer's perspective, NUBOMEDIA capabilities are accessed through a set of **APIs and SDKs**
  - NUBOMEDIA applications can be deployed using the NUBOMEDIA **PaaS Manager**

1. Introduction

References

- Home page

  http://www.nubomedia.eu/

- Developers guide

  http://nubomedia.readthedocs.io/

- GitHub organization

  https://github.com/nubomedia/

- Support for developers

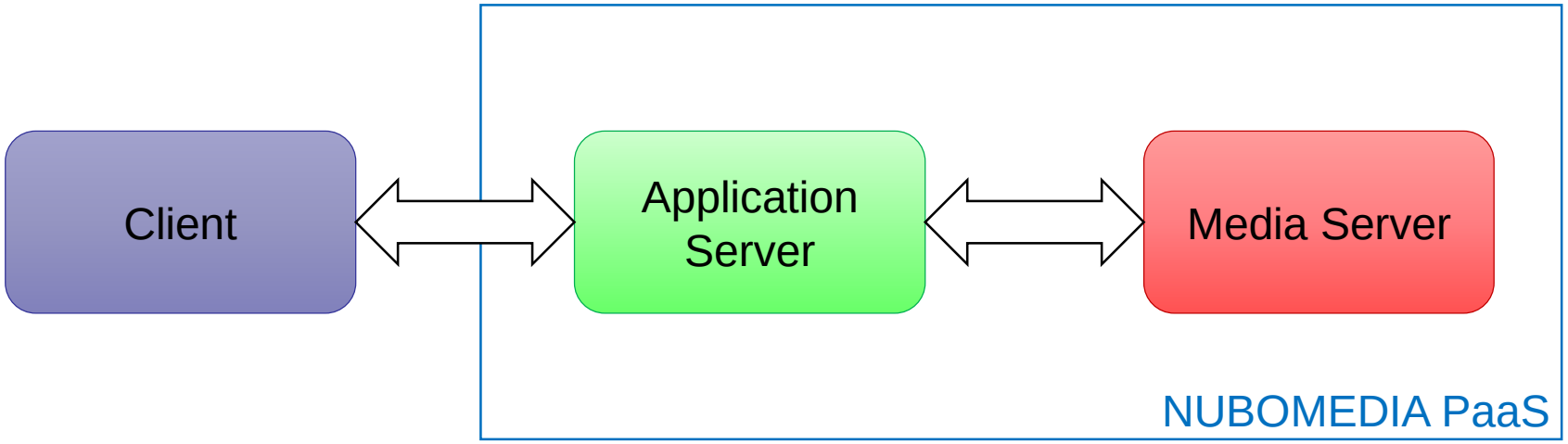
  https://groups.google.com/forum/#!forum/nubomedia-dev

Table of contents

2. NUBOMEDIA overview

Architecture

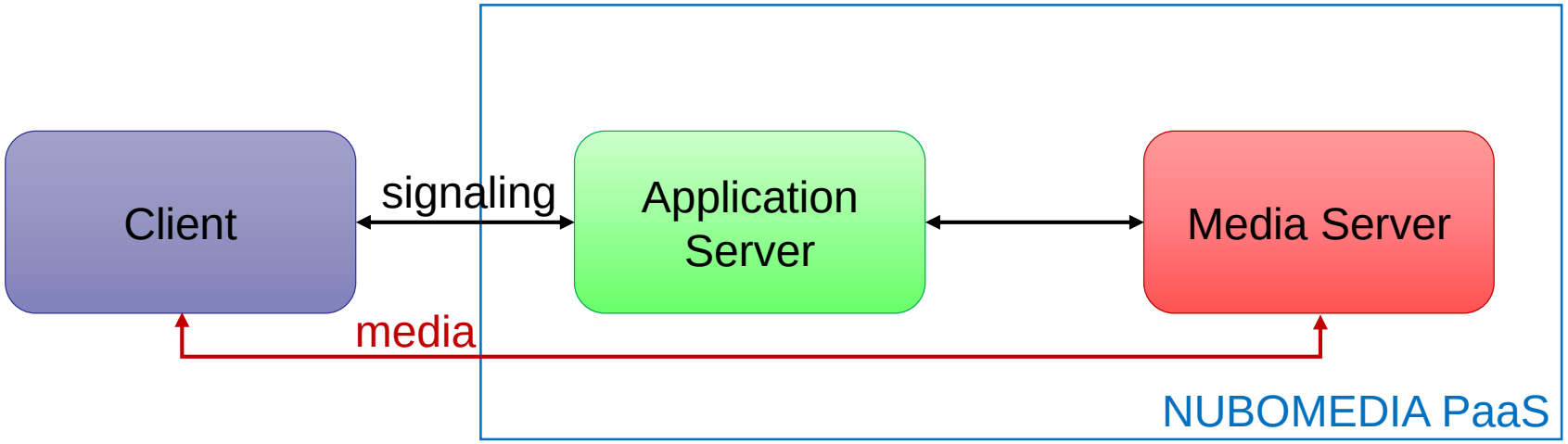
- The Architecture of NUBOMEDIA application follows a three-tier model (inspired in the Web)



2. NUBOMEDIA overview

Architecture

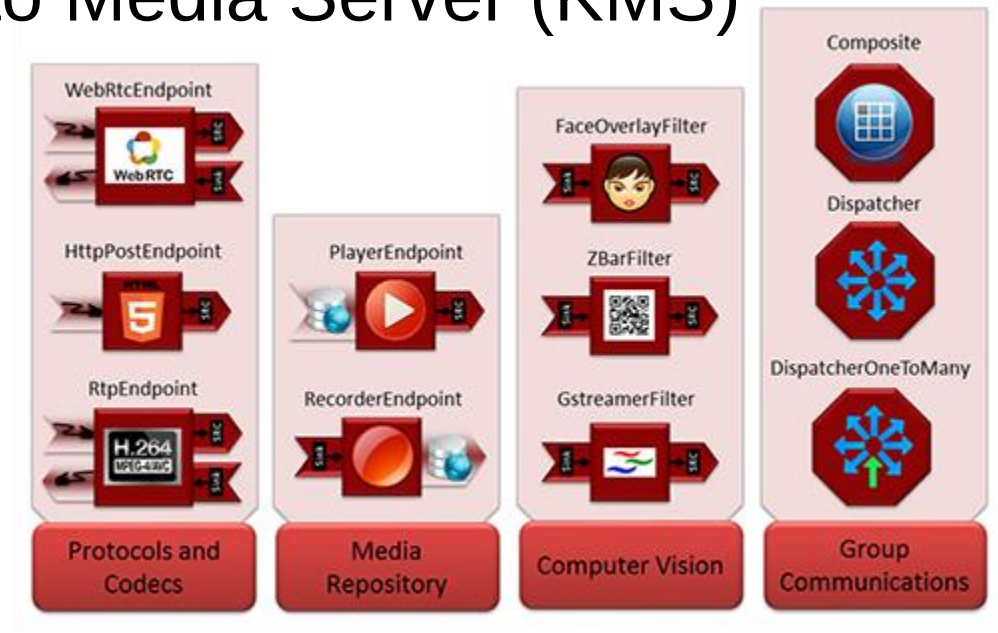
- Like every application with media capabilities, it is important to distinguish between the **media** and **signaling** plane



2. NUBOMEDIA overview

Media API

- NUBOMEDIA Media API allows to **Java** developers consume the media services provided by Kurento Media Server (KMS)
- Concepts:
  - Media Element
  - Media Pipeline



2. NUBOMEDIA overview

Media API

- KMS instances are provided elastically by NUBOMEDIA
  - The number of available KMS instances depends on the PaaS Manager configuration
- Each KMS has a total amount of available points to create Media Pipelines and Media Elements
  - The total points depends on the number of VCPUs of the KMS
  - The type of the instance can be selected on the PaaS Manager configuration

Instance type	# VCPUs	KMS points
Medium	2	200
Large	4	400

2. NUBOMEDIA overview

Media API

- Each KMS is controlled by an instance of `KurentoClient`

```
<dependency>
    <groupId>org.kurento</groupId>
    <artifactId>kurento-client</artifactId>
</dependency>

<dependency>
    <groupId>de.fhg.fokus.nubomedia</groupId>
    <artifactId>nubomedia-media-client</artifactId>
</dependency>
```

Dependencies (Maven)

- With each media session an instance of `KurentoClient` should be created

```
KurentoClient kurentoClient = KurentoClient.create();
```

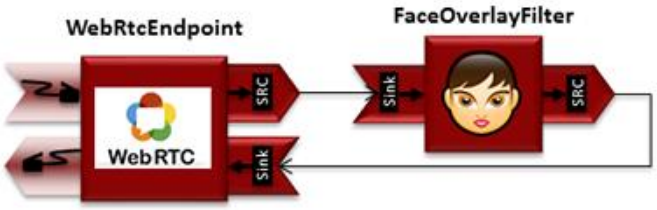
- The number of available points per KMS decreases with each Media Element creation (scaling in/out)

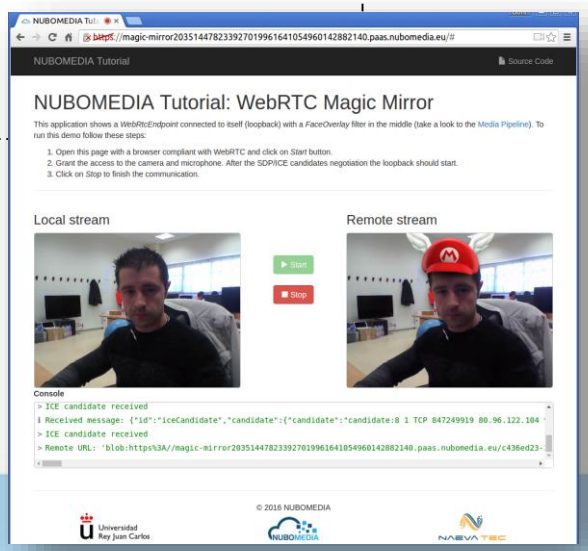
2. NUBOMEDIA overview

Media API

- Example: nubomedia-magic-mirror

```
 // One KurentoClient instance per session
KurentoClient kurentoClient = KurentoClient.create();

// Media logic (pipeline and media elements connectivity)
MediaPipeline mediaPipeline = kurentoClient.createMediaPipeline();

WebRtcEndpoint webRtcEndpoint = new WebRtcEndpoint.Builder(mediaPipeline).build();
FaceOverlayFilter faceOverlayFilter = new FaceOverlayFilter.Builder(mediaPipeline).build();
faceOverlayFilter.setOverlayedImage("http://files.kurento.org/img/mario-wings.png", -0.35F,
        -1.2F, 1.6F, 1.6F);

webRtcEndpoint.connect(faceOverlayFilter);
faceOverlayFilter.connect(webRtcEndpoint);
```





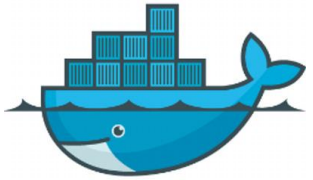
2. NUBOMEDIA overview

PaaS Manager

- The NUBOMEDIA PaaS manager is a tool aimed to control the way in which the NUBOMEDIA applications are built and deployed inside the NUBOMEDIA PaaS
- The capabilities provided by the Paas Manager can be used by developers using the **PaaS GUI**:
  - The PaaS Manager GUI is a web application that allows to use the NUBOMEDIA PaaS Manager

2. NUBOMEDIA overview

PaaS Manager



- Internally, the NUBOMEDIA PaaS uses **Docker containers** to deploy applications
- Therefore it is a requirement to include a **Dockerfile** in GitHub repository to be deployed on NUBOMEDIA
- Example:

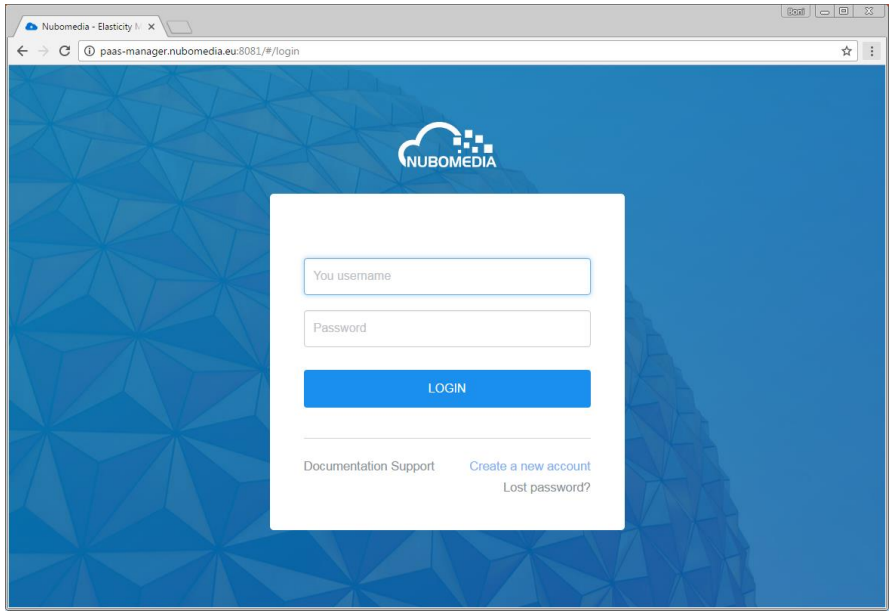
```
FROM nubomedia/apps-baseimage:src

MAINTAINER Nubomedia

ADD . /home/nubomedia

ENTRYPOINT cd /home/nubomedia && mvn spring-boot:run
```

https://docs.docker.com/engine/reference/builder/

2. NUBOMEDIA overview

PaaS Manager

- The PaaS Maanger GUI is a web application to manage NUBOMEDIA applications

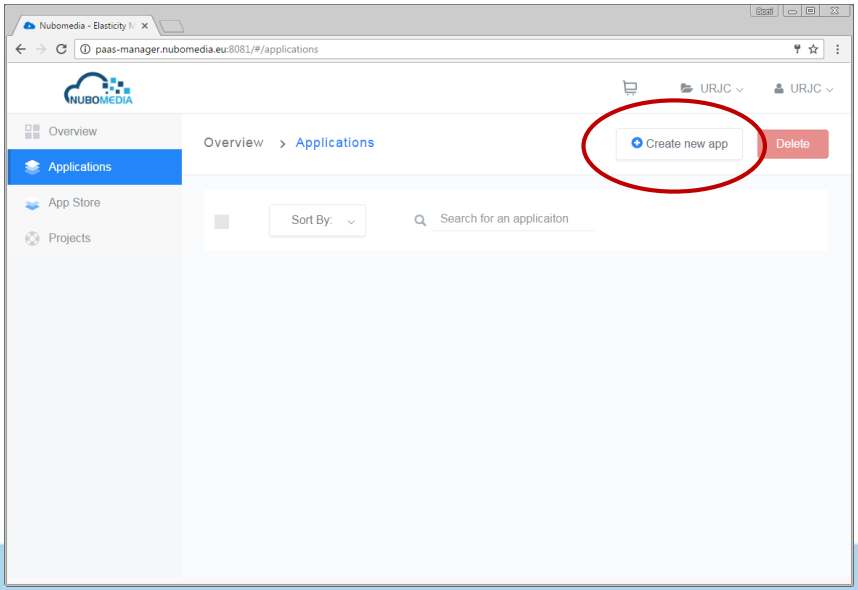


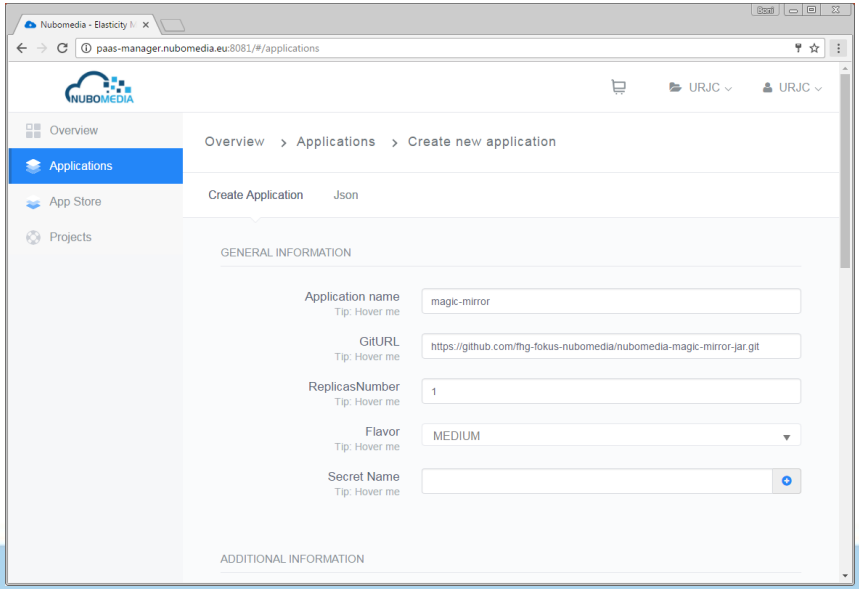
http://paas-manager.nubomedia.eu:8081/

2. NUBOMEDIA overview

PaaS Manager

- A NUBOMEDIA application can be deployed using the PaaS GUI
- It is done providing the GitHub repository URL and a set of configuration parameters





2. NUBOMEDIA overview

PaaS Manager

- Most important configuration values:

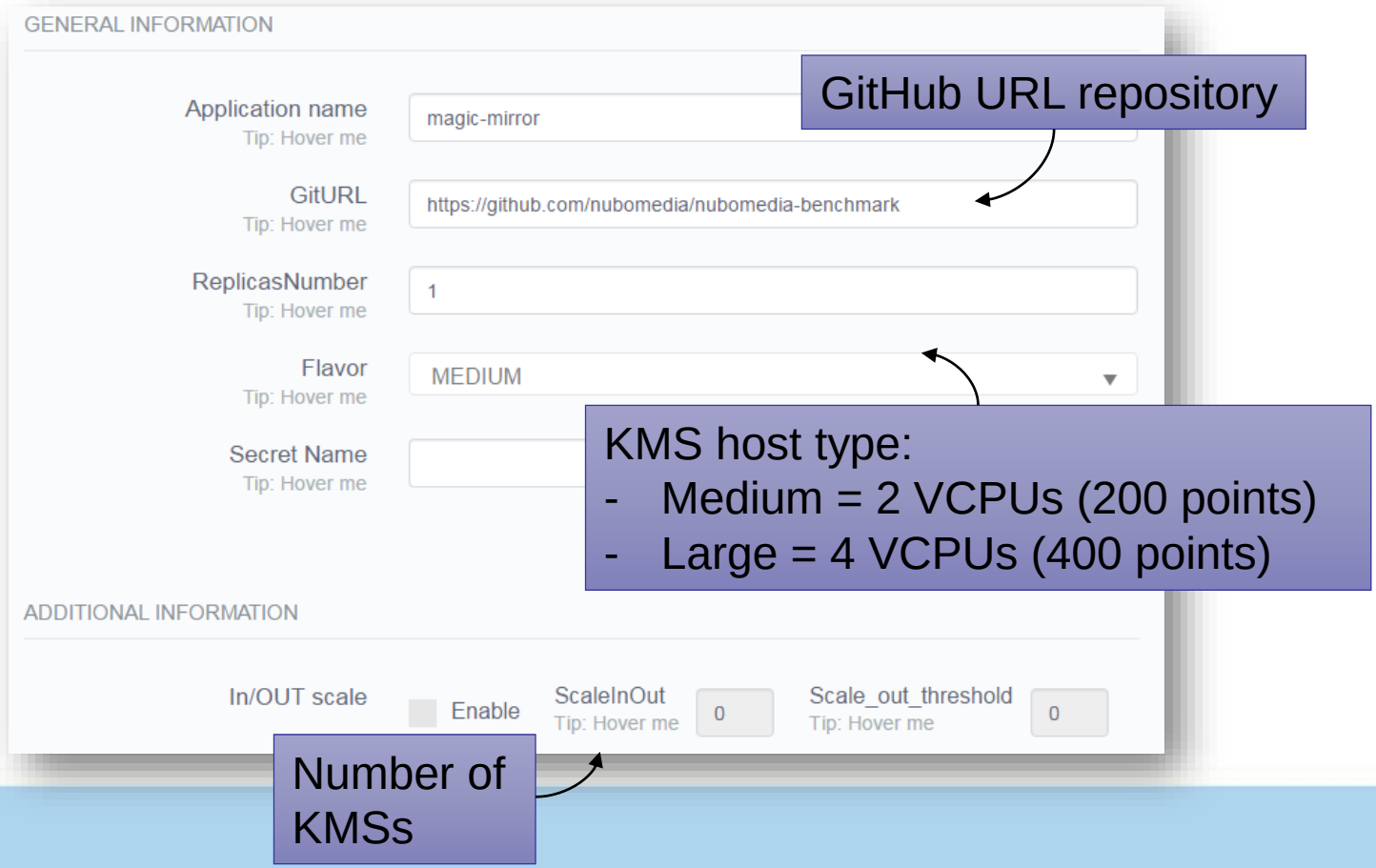


Table of contents

3. NUBOMEDIA Architecture

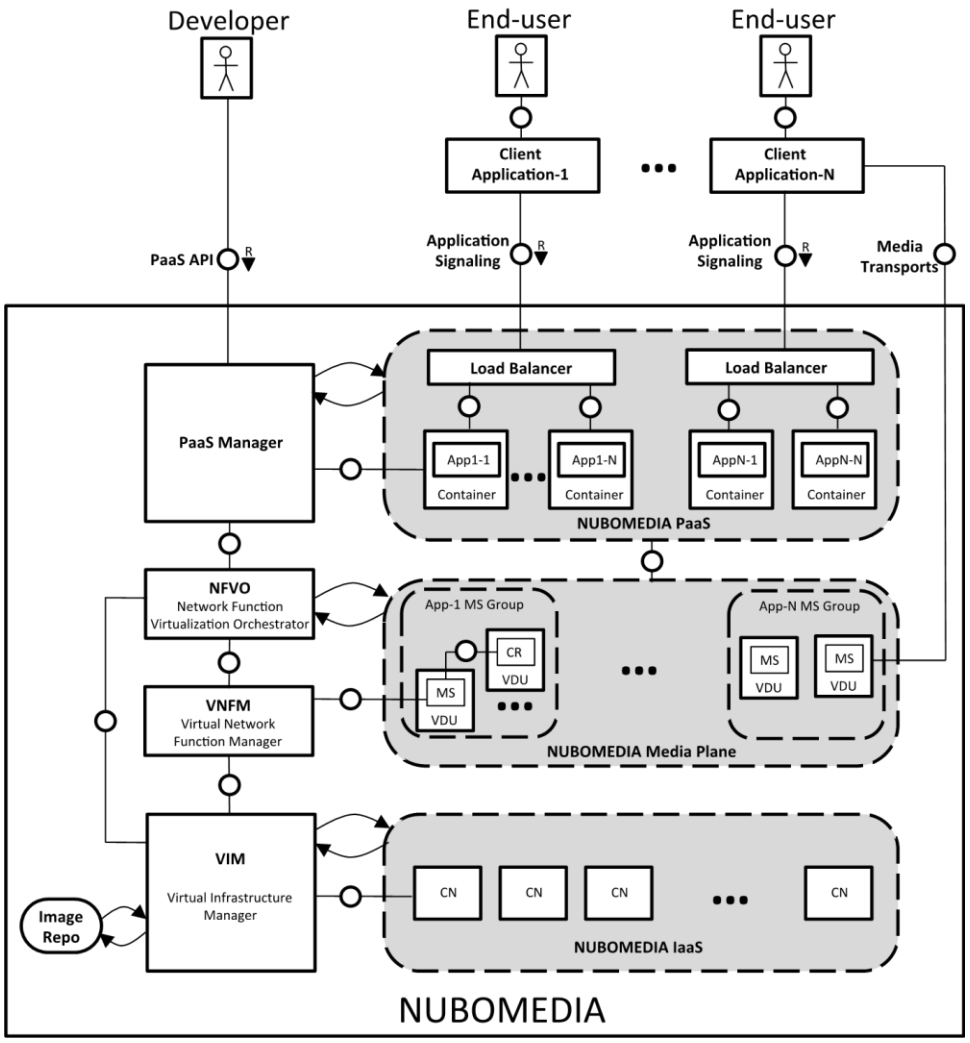


Table of contents

4. Evaluation

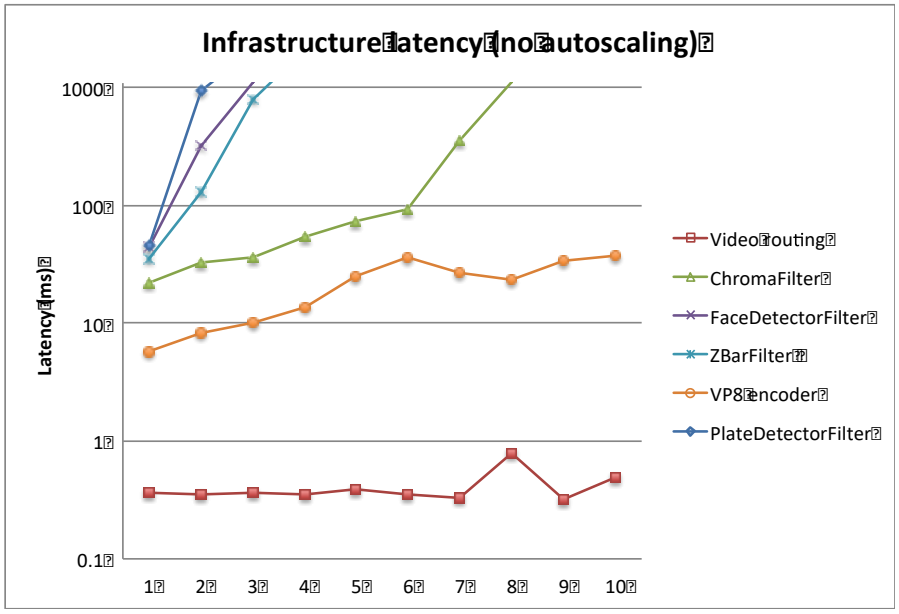
Experiment description

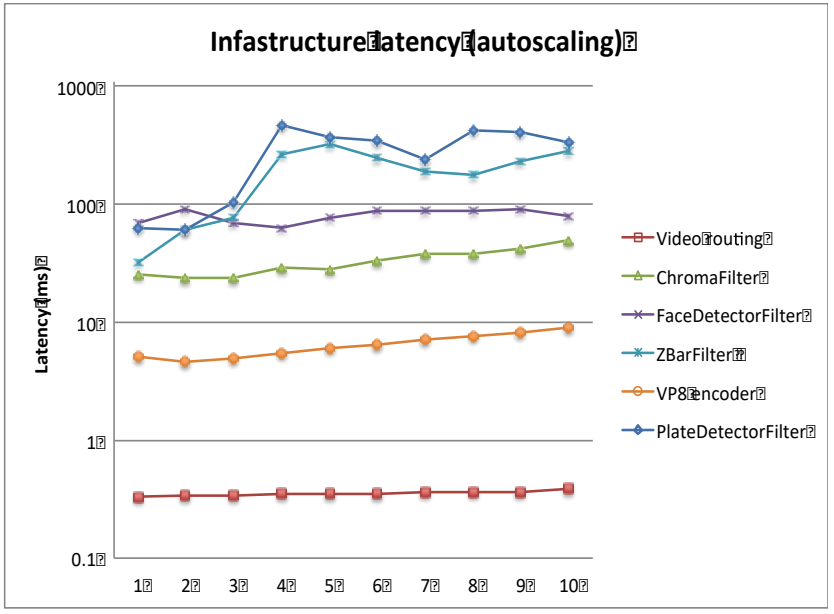
- WebRTC loopback pipeline with different types of filters: encoding, VCA, AR, etc.
- The experiment has been carried out with and without scaling mechanisms
- Data gathered:
  - Media pipeline latency
  - CPU consumption in media server instances

4. Evaluation

Results

– Media pipeline latency

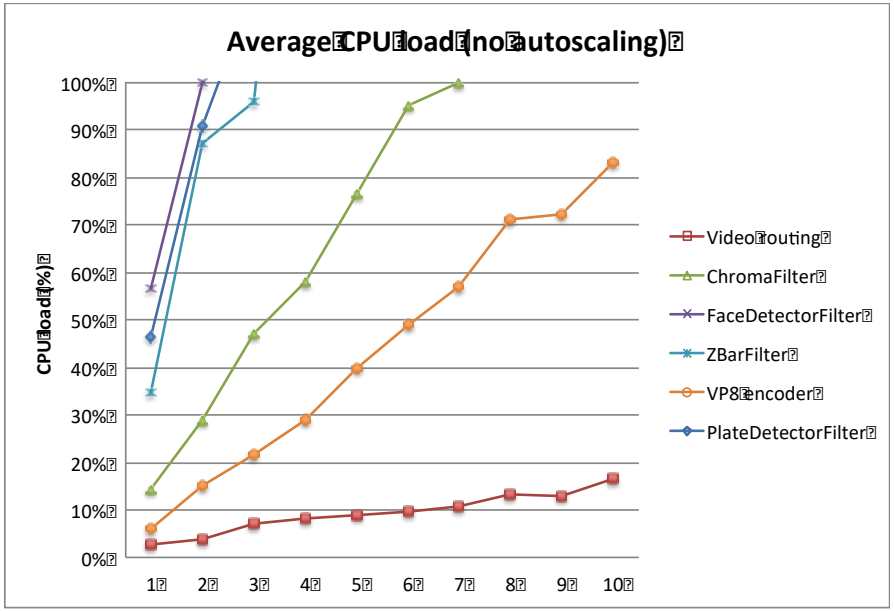




4. Evaluation

Results

– CPU consumption in media server



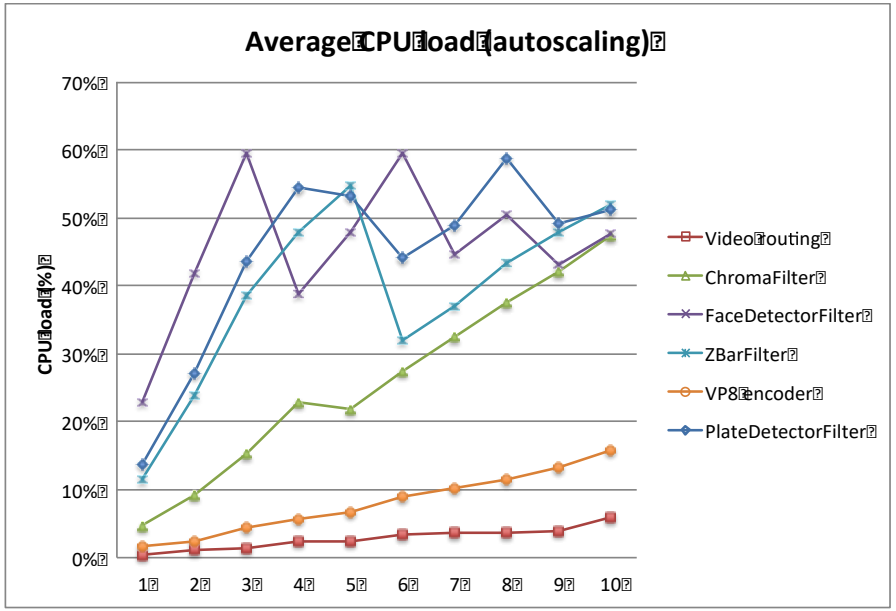


Table of contents

5. Conclusions

- NUBOMEDIA is a PaaS platform enabling the convergence of RTC and multimedia big data through advanced media processing
- It can be used by developers for saving tons of effort when creating such types of applications
- Possible future work: improvement on scheduling and placement algorithms for sessions based on policies beyond the points mechanisms

Thank you!
QA

Boni García
Universidad Rey Juan Carlos (Spain)
boni.garcia@urjc.es