# Locality-aware Cooperation in Distributed IaaS Infrastructures

Jonathan Pastor, Marin Bertier, Flavien Quesnel, Adrien Lebre, Cedric Tedeschi

Departement of Computer Science, Ecole des Mines de Nantes
http://http://www.mines-nantes.fr/

**Abstract.** With the adoption of distributed cloud computing infrastructures as the new platform to deliver utility computing paradigm, new algorithms leveraging peer to peer approaches have been proposed for scheduling virtual machine (VM). Although these proposals considerably improve the scalability enabling the management of hundreds of thousands of VM upon thousands of physical machines (PMs), multisite infrastructures introduce network overhead, which can have a dramatic impact on performances when there is no mechanism in charge of favoring intra-site vs. inter site manipulations.
To reduce such an impact, locality properties should be considered as a key element, e.g. PMs should collaborate first with their neighbourghood from the same geographical site before contacing remote ones. As network bandwidth/latency fluctuate over time, using a static partitionning of the resources is not enough.
This paper introduces a new building block built on a vivaldi overlay that maximizes efficient collaborations between PMs. We combined this mechanism with DVMS , a large scale virtual machine scheduler and show its benefit by discussing several experiments performed on four distinct sites of the Grid'5000 tesbed. Thanks to our proposal and without changing the scheduling decision algorithm, the number of inter-site operations has been reduced by 66properties to improve performance of massive distributed cloud platforms.

**Keywords:** Cloud computing, locality, peer to peer, network overlay, vivaldi, DVMS, virtual machine scheduling

## 1 Introduction

Introduced few years ago [**?**], the new trend to deliver cloud computing resources, in particular Infrastructure as a Service solutions, consists in leveraging several infrastructures distributed world-wide. If such distributed cloud computing platforms deliver undeniable advantages to address important challenges such as reliability, latency or even in somehow jurisdiction concerns, most mechanisms that were previously used to operate centralized IaaS platforms must be revisited to offer the same level of transparency for the end-users. Keeping such an objective in mind, the use of P2P paradigm is strongly investigated. This is particularly

true for instance for scheduling algorithms in charge of assigning VMs on top of PMs according to their effective needs (and reciprocally usages). Indeed and although major improvements have been done, centralized approaches [**?**] are not scalable enough and hierarchical solutions [**?**] face important limitations regarding the reactivity to take into account physical topology changes, an important criteria in such widely distributed infrastructures.

# 2 Background

## 2.1 DVMS

- fundamentals
- limitations

## 2.2 P2P - locality

- Vivaldi
- active/lazy clustering

# 3 Contributions

## 3.1 Locality based overlay

- clustering
- Vivaldi + spirale

## 3.2 Dvms + PeerActor + locality

# 4 Experimentations

## 4.1 Implementation

A prototype of DVMS leveraging locality based overlay has been developed. The current version of DVMS are been developed over the PeerActor abstraction. PeerActor provides network abstraction that enables the design of distributed algorithm that are network overlay agnostic. We have developed two different overlay for the Peer Actor abstraction: Chord and a locality based overlay over Vivaldi.

The strength of this software architecture is that to enable an algorithm (like DVMS) to comply with a given network overlay, it only have to follow the Peer Actor API. This way, we were able to run DVMS over Chord or Vivaldi without any modification in its source code.

### 4.2 Grid5000' experiments

**Objectives** The prototype has been tested with a various number of experiments conducted on the Grid5000' testbed. The main objective of the experiments was to estimate impact of locality on the performance of a distributed scheduling algorithm. A significant portion of the reconfiguration time is spent in live migration of virtual machines, which depends of network parameters such as latency and bandwidth. One way to improve performance of distributed scheduling algorithm is is to promote collaboration between close ressources, which can be reach by maximising this ratio:

$$\frac{number\ of\ intrasite\ migrations}{number\ of\ migrations}$$

**Experimental protocol** For each experiment, we booked 40 compute servers spread on 4 geographical sites and 1 service server. The compute servers were used to run virtual machines and DVMS while the service node is used to stress several parameters of virtual machines.

Each compute node will host a number of virtual machines proportional to the number of CPU cores it has. In our case:

$$number\ of\ virtual\ machines\ =\ 1.3\ \times\ number\ of\ cores$$

**Results** The impact of locality on DVMS is significant: using a Vivaldi based network overlay leads to an average number of 83% of intrasite migrations while using a Chord based DVMS leads a ratio of 50% of intrasite migrations, as depicted in the following table:

| network overlay | average number of intrasite migrations | average number of migrations |
|---|---|---|
| Vivaldi | 83 | 100 |
| Chord | 40 | 80 |

Complete this section with more experimentation results.

## 5 Related work

### 5.1 DVMS

### 5.2 P2P

## 6 Conclusion