Entropy: a Consolidation Manager for Clusters

Fabien Hermenier[1] Xavier Lorca[2] Jean-Marc Menaud[1] Gilles Muller[3]
Julia Lawall[4]

[1] ASCOLA group, École des Mines de Nantes
[2] Constraints group, École des Mines de Nantes
[3] INRIA-Régal, École des Mines de Nantes
[4] DIKU, University of Copenhagen

International Conference on Virtual Execution Environments,
Washington D.C., March 12 2009

Context

Cluster environment

- Static allocation of the resources to jobs
- Resources are underused
- Static allocation of CPUs vs. dynamic utilization

Dynamic Consolidation

- Each task of a job is embedded into a Virtual Machine (VM)
- Resources are allocated depending on tasks needs
- VMs are packed to be hosted on a reduced number of nodes
- VMs are re-packed when necessary with migrations

Challenge

Issues

- Packing the VMs may require several migrations
- Some migrations have to be delayed to succeed.
- Temporary hosting may be necessary

$\longrightarrow$ Migrations take time
$\longrightarrow$ Performance degrades

Reactivity is essential

Our proposal

Entropy

- A dynamic consolidation manager for clusters,
- Plans the migration process
- Reduces the duration of the migration process to improve reactivity

1 Design

2 Packing the Virtual Machines

3 Planning the migrations

4 Minimizing the migrations

5 Evaluation

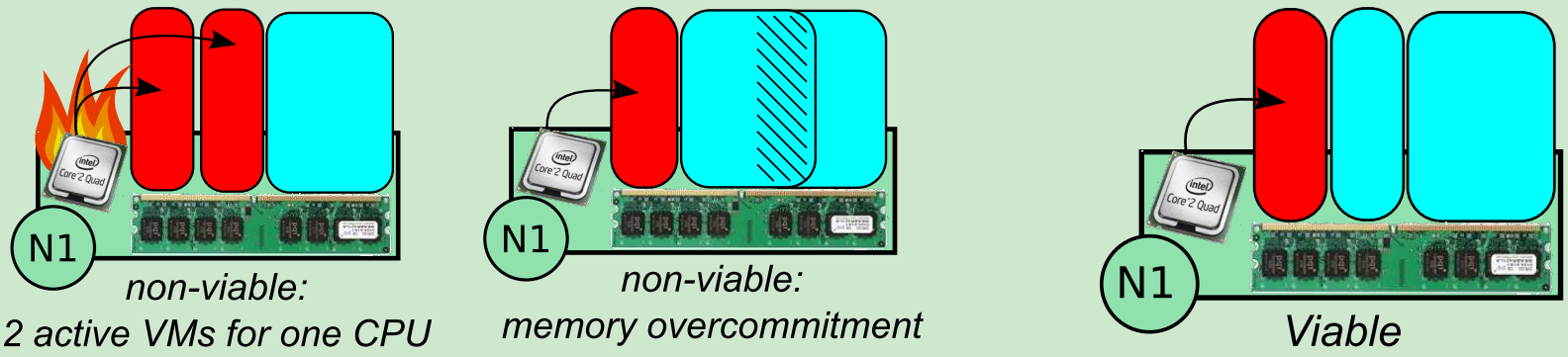
6 Conclusion

Global Design of Entropy

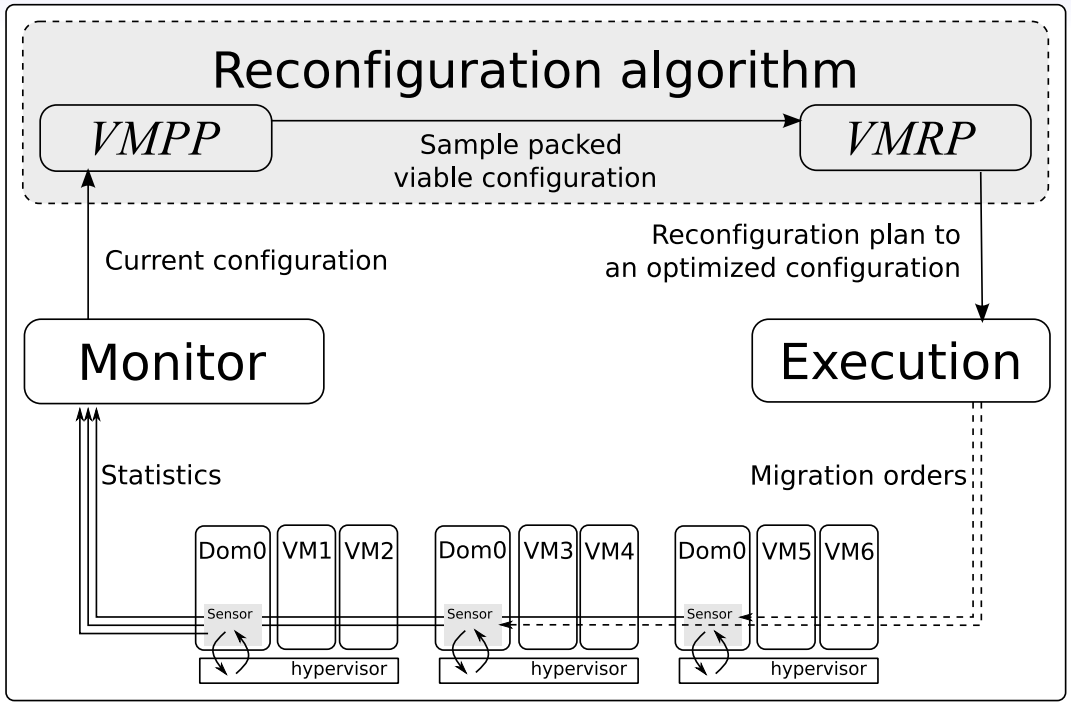
A Configuration :

- Each VM is assigned on a node,
- Each VM requires a fixed amount of memory.
- VMs executing a computation are active and require a private CPU.
- May be viable

Example



Global Design of Entropy

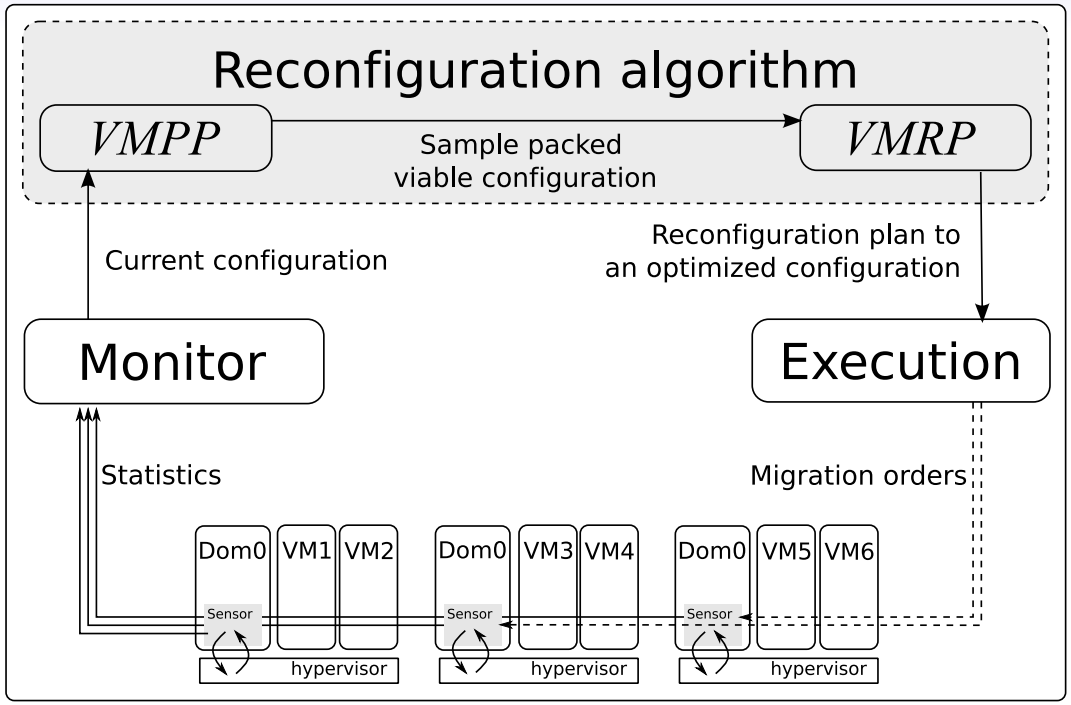


Monitor

Extract the current configuration :

- The position of each VMs and its CPU consumption
- An indication of which of the VMs are active and inactive

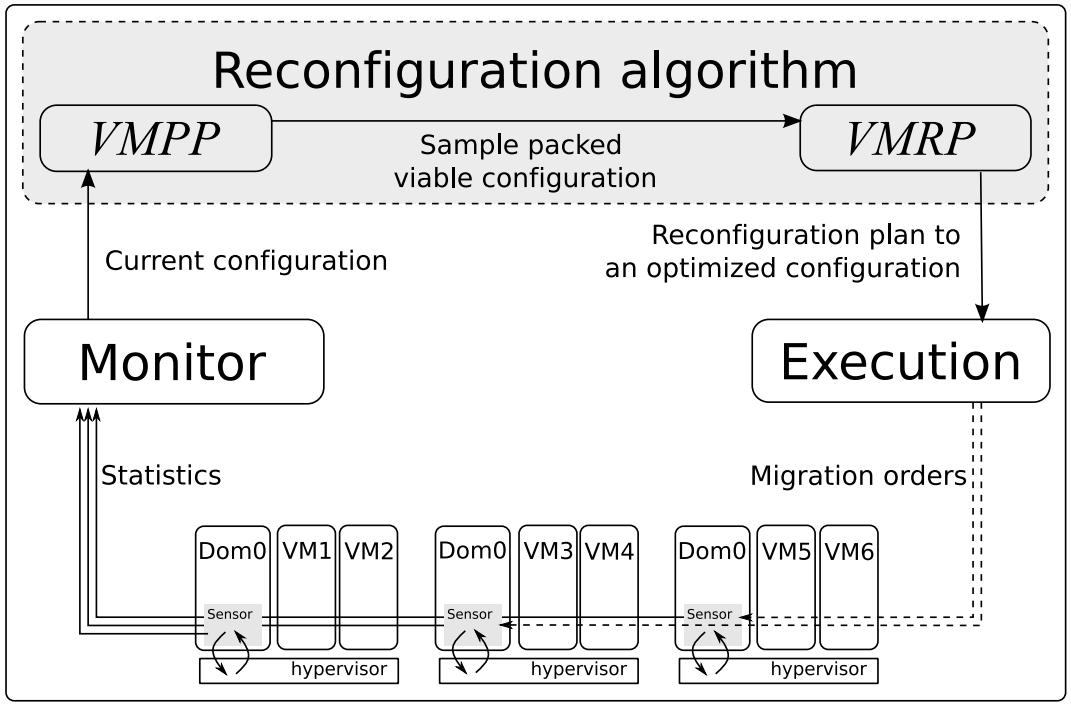
Global Design of Entropy



Reconfiguration Algorithm

- **VMPP** - Compute a viable configuration using a minimum number of nodes
- **VMRP** - Plan and reduce the reconfiguration process if necessary

Global Design of Entropy



Execution

- Decompose a plan into simple migrations
- Migrations orders are sent to the concerned VMM

1 Design

2 **Packing the Virtual Machines**

3 Planning the migrations

4 Minimizing the migrations

5 Evaluation

6 Conclusion

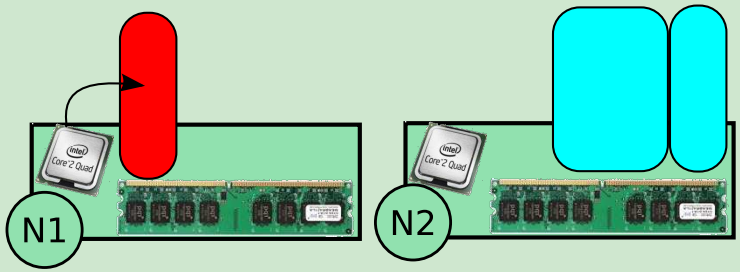
Packing the Virtual Machines

Definition

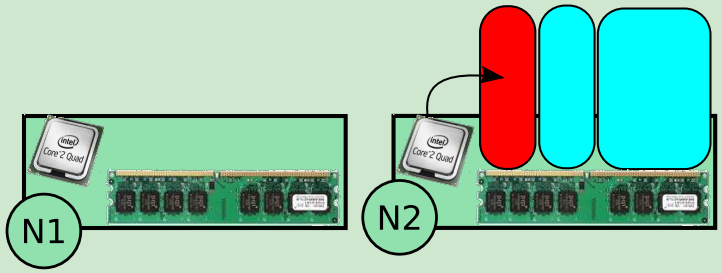
The Virtual Machines Packing Problem (VMPP)

- Compute the minimum number of nodes needed for a viable configuration

Example



(a) viable but non minimal



(b) viable and minimal

Packing the virtual machines

Approach

- Based on constraint programming,
- Each condition defining a viable configuration is a constraint.

The constraint solver :

- Computes a viable configuration from the current one
- Reduces the number of used nodes until the minimum or a timeout.

1 Design

2 Packing the Virtual Machines

3 Planning the migrations

4 Minimizing the migrations

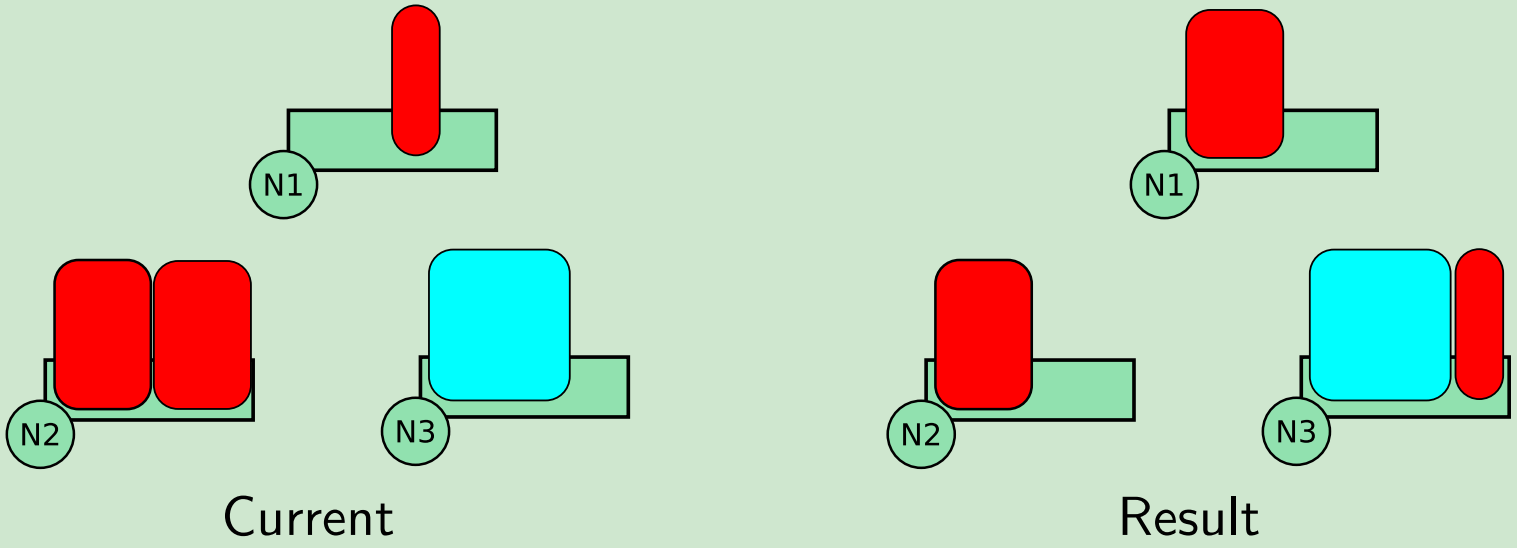
5 Evaluation

6 Conclusion

Planning a reconfiguration

Migrations have to be ordered

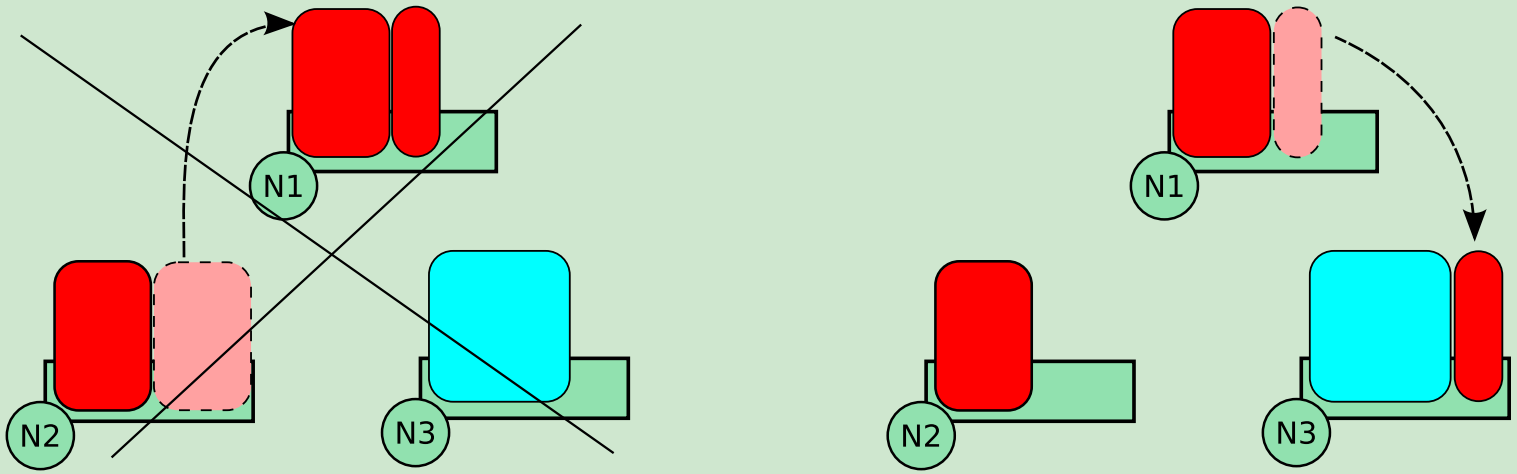
Example



Planning a reconfiguration

Migrations have to be ordered

Example

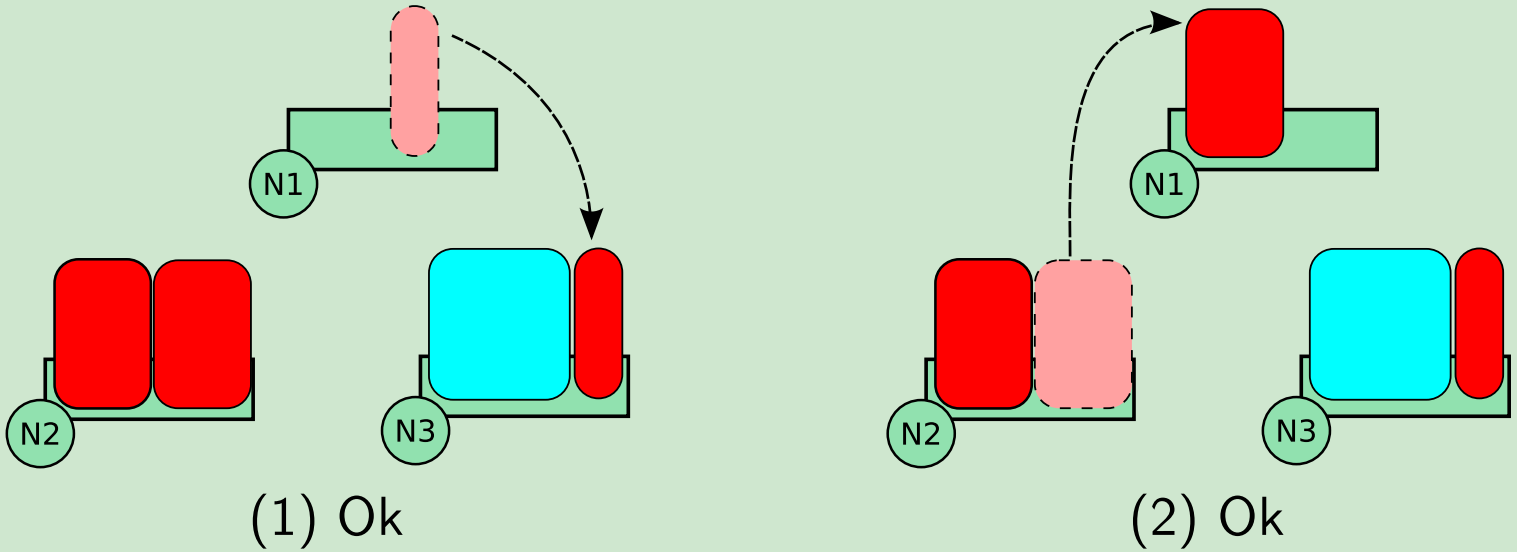


(1) non-viable (2)

Planning a reconfiguration

Migrations have to be ordered

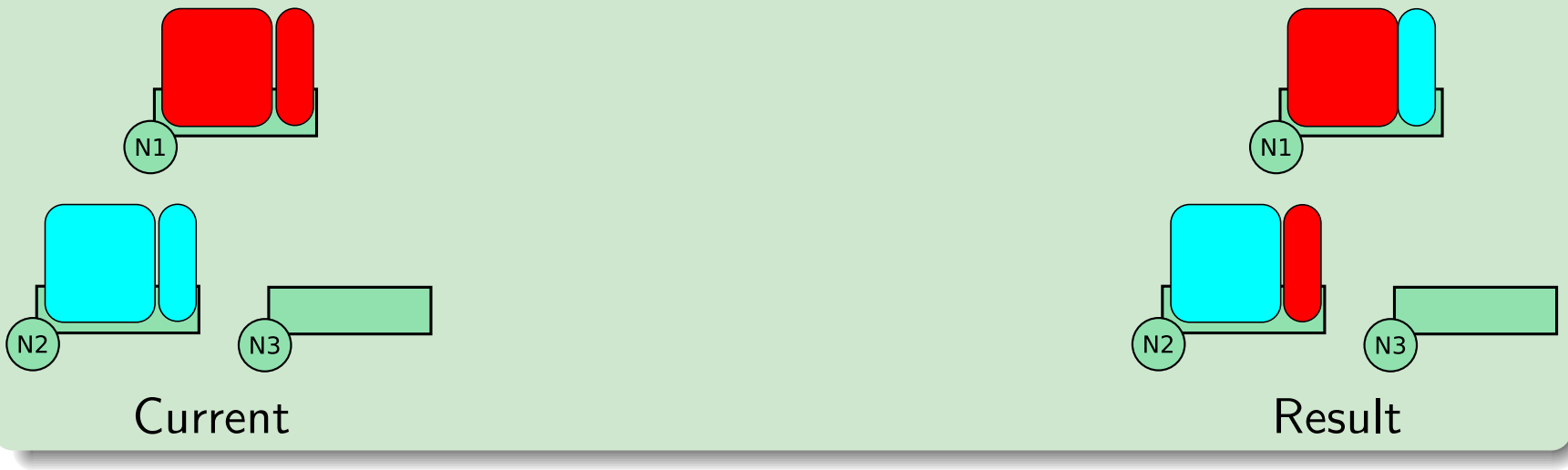
Example



Planning a reconfiguration

Inter-dependant migrations require a pivot

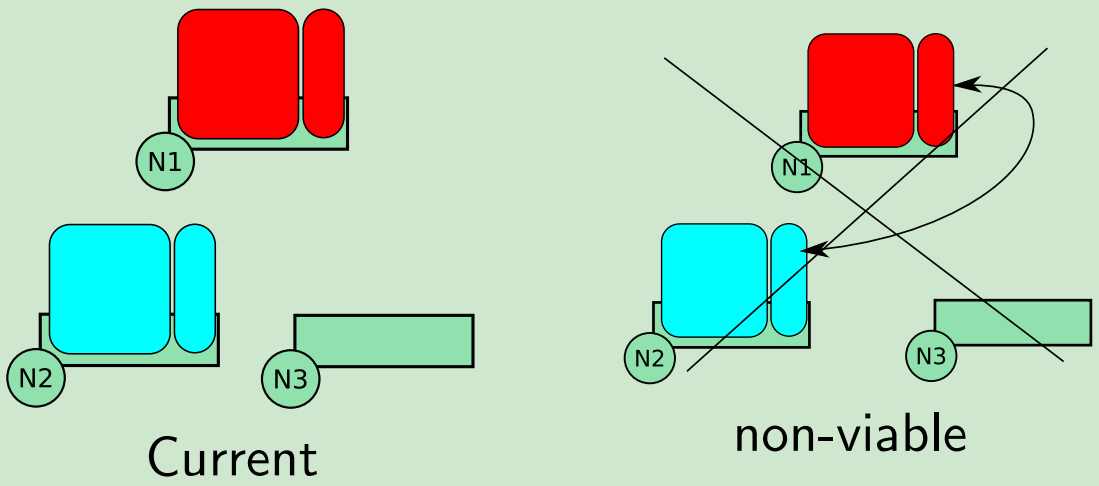
Example



Planning a reconfiguration

Inter-dependant migrations require a pivot

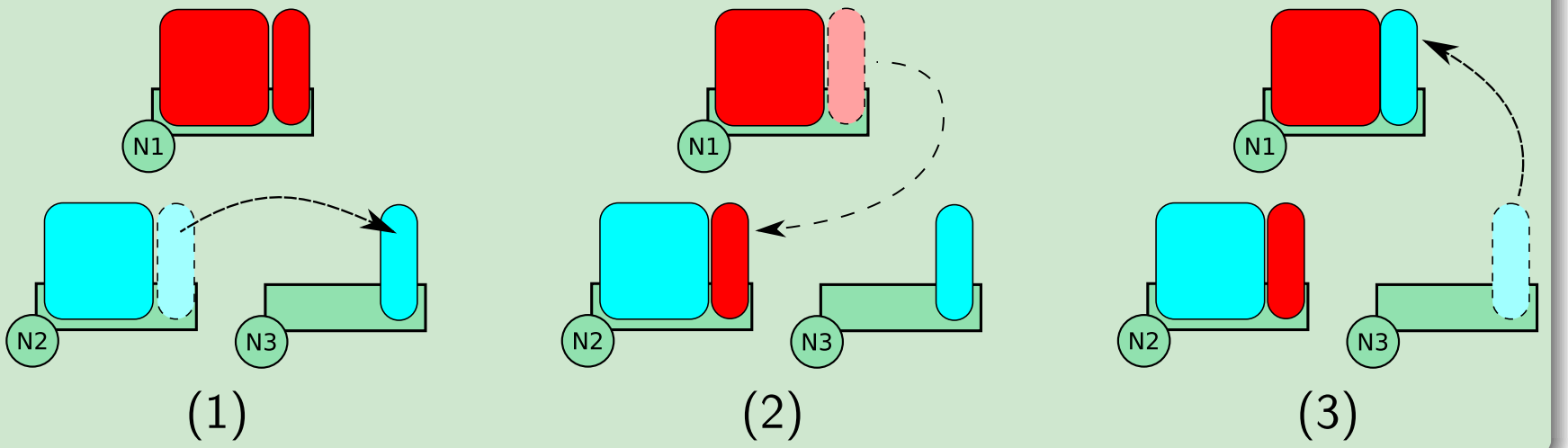
Example



Planning a reconfiguration

Inter-dependant migrations require a pivot

Example



Planning a reconfiguration

The Reconfiguration Plan

- Describes a viable reconfiguration process
- Migrations feasible in parallel are grouped into a step
- Steps are executed sequentially

1 Design

2 Packing the Virtual Machines

3 Planning the migrations

4 Minimizing the migrations

5 Evaluation

6 Conclusion

Reducing the reconfiguration process

VMRP - Looking for an equivalent configuration

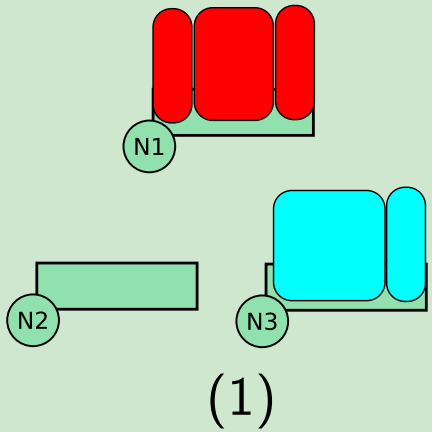
- Which is a solution of the VMPP
- Where its associated plan has a minimal "cost"

Method

- The cost of a plan is estimated using a migration cost model
- The VMRP computes equivalent configurations with "cheap" reconfiguration plans until the minimum or a timeout.

Reducing the reconfiguration process

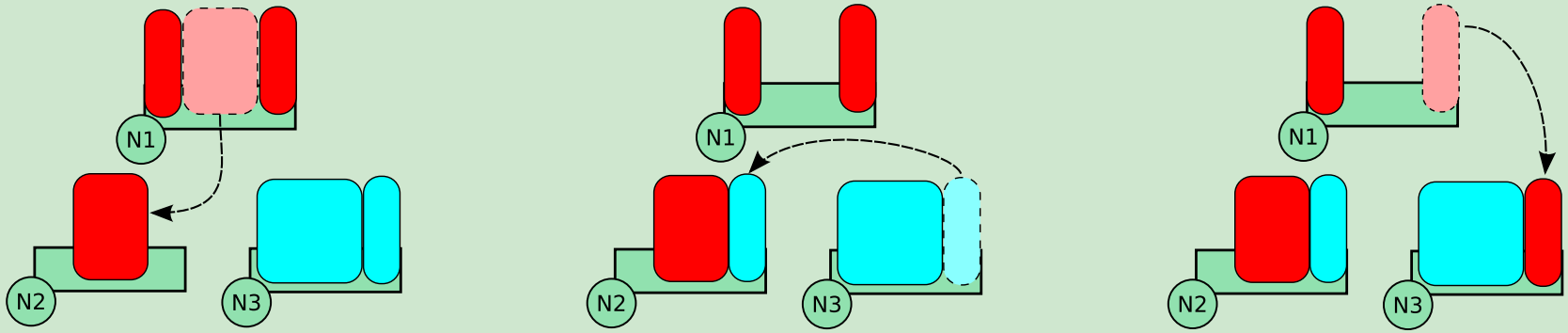
Example



(1)

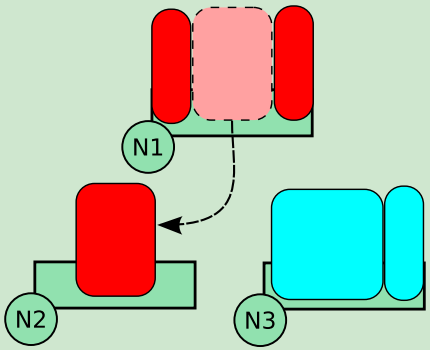
Reducing the reconfiguration process

Example

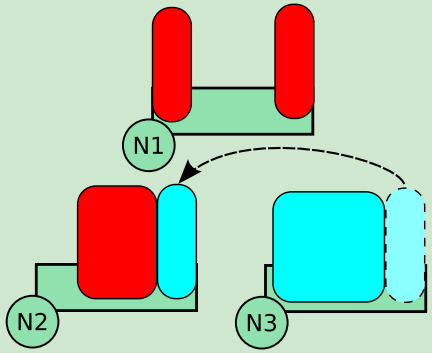


Reducing the reconfiguration process

Example

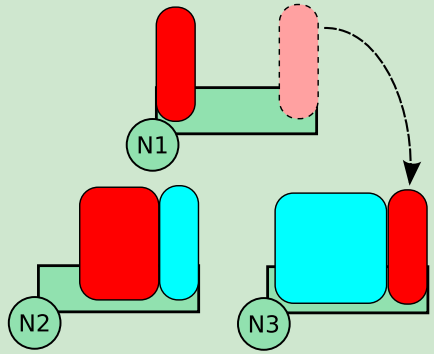


(1)

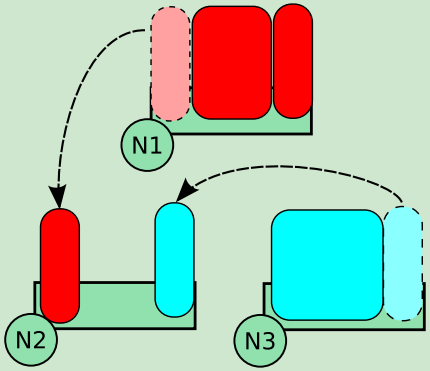


(2)

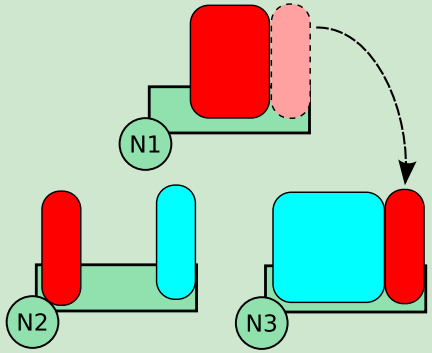
$cost = 9$



(3)



(1)



(2)

$cost = 4$

1 Design

2 Packing the Virtual Machines

3 Planning the migrations

4 Minimizing the migrations

5 **Evaluation**

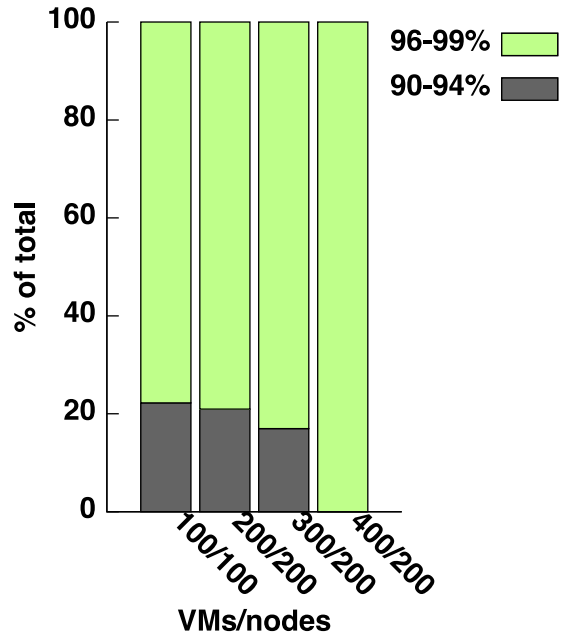
6 Conclusion

Comparison against "First Fit Decrease"

Timeouts to have a non-trivial solution with Entropy estimated using random configurations :

- 30 secs. for the packing
- 35 secs. for minimizing the migrations

- The packing is equivalent or better. Small benefits for 42% of the configurations
- Cost of the resulting plan reduced by at least 90%



Cost

Experiments on a cluster

Environment

- 1 node hosting the consolidation manager
- 3 nodes for serving the VMs virtual disks
- 35 nodes running a hypervisor
- 35 VMs executing a collection of NASGrid Benchmarks

Method

- All the benchmarks are launched at the same time
- Comparison between
  - Static allocation without consolidation
  - Dynamic consolidation using FFD
  - Dynamic consolidation using Entropy

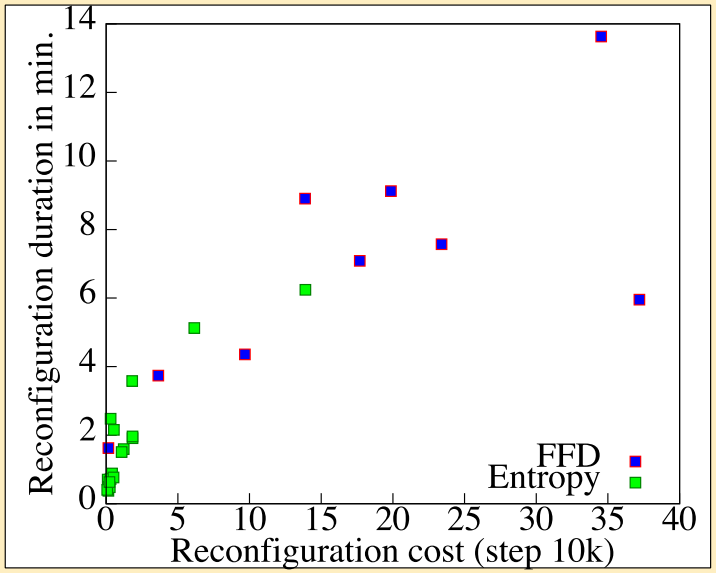
Experiments on a cluster

Benefits

- Better reactivity
- Stable packing
- Reduced overhead

Comparing the reconfigurations against FFD

- Cost : -90%
- Duration : -74%
- Nb of reconfigurations : x2

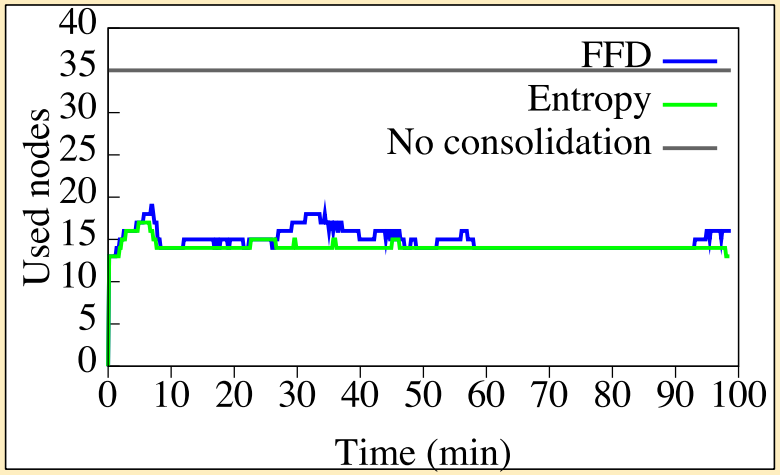


Experiments on a cluster

Benefits

- Better reactivity
- Stable packing
- Reduced overhead

Impact on the packing



Smaller plans imply fewer pivots

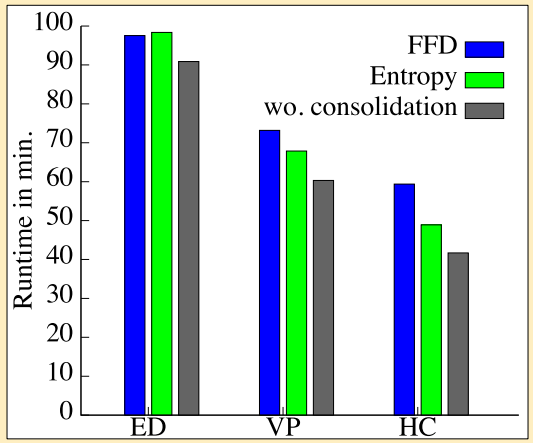
Experiments on a cluster

Benefits

- Better reactivity
- Stable packing
- Reduced overhead

Impact on performance

- Overhead reduced by 9%
- Node per hour consumption reduced by 25%



1 Design

2 Packing the Virtual Machines

3 Planning the migrations

4 Minimizing the migrations

5 Evaluation

6 Conclusion

Conclusion

The interest of the dynamic consolidation is limited by the duration of the reconfiguration process.

Entropy

- Reducing the cost of a plan is an efficient solution to reduce its duration
- 1 minute to compute a solution reduces the reconfiguration process by up to 8 minutes.
- Reduces the nodes per hour consumption by 25% as compared to FFD and the overhead by 9%.

Questions ?

http ://entropy.gforge.inria.fr

- Binary and sources available on LGPL
- Uses the Xen Hypervisor and the ganglia monitoring system