

Mathieu Bivert, CSSR, bivert@essi.fr

PFE: Cahier des charges

Placement constraints for a better QoS in clouds



Entreprise INRIA

Lieu Sophia-Antipolis, France

Responsable Fabien Hermenier, équipe OASIS, fabien.hermenier@unice.fr

Résumé

Blablabla français

Abstract

Blablabla english

Contents

1	Description du projet	3
1.1	Contexte de travail	3
1.2	Motivations	3
1.3	Défis	3
1.4	Objectifs	3
1.5	Scénarios	4
1.6	Critère de succès	4
2	État de l'art	4
2.1	Description générale	4
2.2	foo	4
2.3	bar	4
3	Méthodologie et planification	4
3.1	Stratégie générale	4
3.2	Découpage en lots	5
3.3	Plannification	5
3.4	Livrables associés au projet	5
3.5	Jalons	5
4	Description de la mise en œuvre du projet	5
4.1	Interdépendance des lots et tâches	5
4.2	Description des lots	5
4.3	Résumé de l'effort	5
4.4	Gestion du risque	5
5	Participants	5
5.1	Mathieu Bivert - CSSR	5
5.2	Fabien Hermenier - OASIS/INRIA	5

1 Description du projet

1.1 Contexte de travail

Les clouds (Amazon EC2, Microsoft Azure, etc.) proposent à leurs clients un accès à une infrastructure informatique. Une entreprise peut ainsi déléguer une grande partie de l'administration et de son matériel sur un cloud, ce qui lui permet d'avoir une garantie quant-à la qualité du service fournis, tout en réduisant les coûts.

Afin de rentabiliser au mieux le matériel, les fournisseurs de clouds utilisent généralement la virtualisation : un même serveur physique peut ainsi héberger plusieurs systèmes virtuels, fournissant des services quelconques (emails, serveurs web, accès à un OS complet, etc.).

1.2 Motivations

La question de la répartition des machines virtuelles sur les machines physiques se pose alors pour des raisons diverses et variées:

maintenance un serveur physique peut tomber en panne, ou nécessiter une réparation, auquel cas les programmes tournant dessus doivent être migré ailleurs, afin de garantir au client une certaine qualité de service (QoS);

sécurité il peut s'avérer risquer pour un programme d'un client traitant des données sensibles (eg. données bancaires) de se retrouver au même endroit qu'un programme d'un autre client;

économie d'énergie où il peut être avantageux de réduire le nombre de serveurs physiques allumés, pour maximiser le rendement des autres machines physiques du cloud;

QoS où, à l'inverse de l'économie d'énergie, il est bon de garder des ressources supplémentaires disponibles immédiatement, de façon a ne pas perdre de temps (et donc en QoS) à redémarrer un autre serveur;

licence les entreprises fournissant les systèmes de virtualisation proposent des licences selon différents critères (eg. nombre de machines virtuelles lancées, utilisation de ressources (CPU, RAM, etc.));

plateforme plusieurs plateformes de virtualisations sont disponibles (eg. Xen, VMWare, Citrix); une autre contrainte sur la répartition des machines virtuelles se pose alors, un serveur physique ne faisant tourner qu'un seul type de plateforme;

...

1.3 Défis

Afin de pouvoir répondre aux besoins exprimés par l'un des domaines cité dans le paragraphe précédent, il est nécessaire de commencer par formaliser le problème. En d'autres termes, donner une définition mathématiques des contraintes impliquées par la problématique choisie, et s'assurer qu'elles sont envisageables en pratique. Finalement, cette représentation abstraite doit être implémentée sous forme de plugin Java pour entropy [FH09] un manager de clusters reposant sur l'algorithme btrplace¹.

1.4 Objectifs

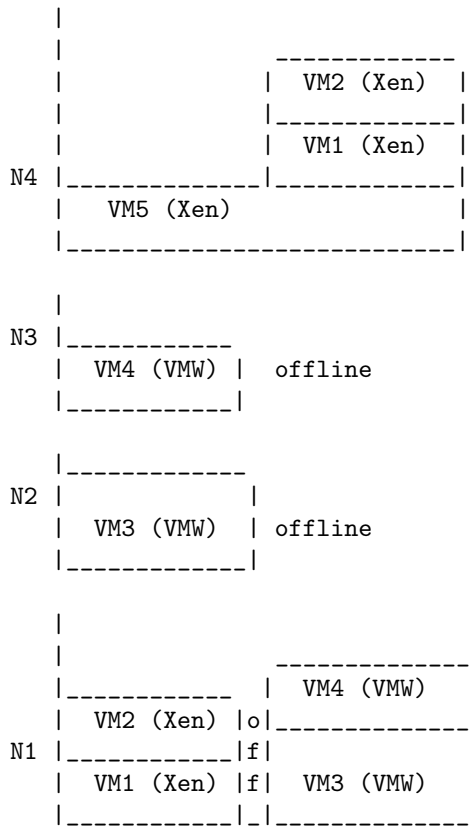
Actuellement, les trois derniers points cités ne sont pas forcément formalisé/implémenté sous une forme satisfaisante. Le projet consiste donc à choisir l'un de ces domaines et à l'ammener vers une forme satisfaisante.

Le dernier point est celui sur lequel se porte ce projet.

¹<http://btrp.inria.fr/sandbox/about.html>

1.5 Scénarios

XXX diagramme.



Sur le diagramme ci dessus, les serveurs physiques N_3 et N_2 doivent être mis hors-ligne pour des questions de maintenances, via la contrainte $offline(N_i)$; ².

Comme aucun serveur VMWare n'est disponible, il est nécessaire de supprimer un serveur Xen, capable d'accueillir VM_4 et VM_3 , par exemple N_1 .

1.6 Critère de succès

bis repetita: trouver un bon formalisme; définir et implémenter les contraintes.

2 État de l'art

2.1 Description générale

2.2 foo

2.3 bar

3 Méthodologie et planification

3.1 Stratégie générale

bis repetita: trouver un bon formalisme; définir et implémenter

²<http://www-sop.inria.fr/members/Fabien.Hermenier/btrpcc/offline.html>

Id	Titre du livrable	Lot(s)	Nature	Date
D_0	Cahier des charges	1	Document	S_4
D_1	Rapport de management	1	Document	S_{20}
D_2	Plugin entropy	1	Logiciel	S_{20}
D_3	Diaporama de présentation finale	1	Document	S_{20}

Id	Jalon de fin de phase	Lot(s)	Date	Vérification
J_0	planification	1	S_4	D_0
J_1	formalisation	1	S_n	D_1 partiel
J_2	implémentation	1	S_{n+k}	D_2, D_1 partiel
J_3	projet	1	S_{20}	D_1, D_2, D_3

3.2 Découpage en lots

bis repetita: trouver un bon formalisme; définir et implémenter

3.3 Plannification

gant

3.4 Livrables associés au projet

3.5 Jalons

4 Description de la mise en œuvre du projet

4.1 Interdépendance des lots et tâches

bis repetita: trouver un bon formalisme; définir et implémenter

4.2 Description des lots

bis repetita: trouver un bon formalisme; définir et implémenter

4.3 Résumé de l'effort

4.4 Gestion du risque

5 Participants

5.1 Mathieu Bivert - CSSR

5.2 Fabien Hermenier - OASIS/INRIA

Références

- [FH09] Jean-Marc Menaud Gilles Muller Julia lawall Fabien Hermenier, Xavier Lorca. Entropy : a consolidation manager for clusters. 2009.