

Mathieu Bivert, CSSR, bivert@essi.fr

PFE: Cahier des charges (DOW)

Placement constraints for a better QoS in clouds



Entreprise Université de Nice-Sophia Antipolis

Lieu Sophia-Antipolis, France

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

1 Vocabulaire et notations

Type entier t associé à chaque système de virtualisation;

VM machine virtuelle, notée $v \in \mathcal{V}$, à laquelle est associée un type fixe $T(v)$ et une place $P(v)$;

Nœud serveur physique, noté $n \in \mathcal{N}$, doté d'un type courant $T(n)$ et d'un ensemble de types possibles \mathcal{T}_n ;

Reconfiguration opération durant laquelle BtrPlace change le placement des VMs sur les nœuds, en fonction des contraintes établies par l'utilisateur;

Slices la modélisation des actions de reconfiguration [FH12] est réalisée à l'aide de *slices*, qui correspondent à une durée finie pendant un processus de reconfiguration, durant laquelle des ressources sont utilisées. On distingue deux types de slices:

consuming slice , $c \in \mathcal{C}$, où les ressources sont utilisées au début de la reconfiguration;

demanding slice , $d \in \mathcal{D}$, où les ressources sont utilisées à la fin de la reconfiguration;

La fonction T associe à une VM ou un nœud son type; la fonction P associe à une VM un nœud.

Un nœud est doté d'une nouvelle dimension de type. Celle-ci est booléenne : soit le type change, auquel cas, la valeur est de 1, sinon, elle vaut 0. Dans les graphes suivants, elle est représentée à part pour des questions de lisibilité.

2 Configuration d'exemple

2.1 Cas général

Dans un premier temps, on cherche à obtenir une configuration minimaliste, mettant en œuvre suffisamment d'éléments pour représenter le problème:

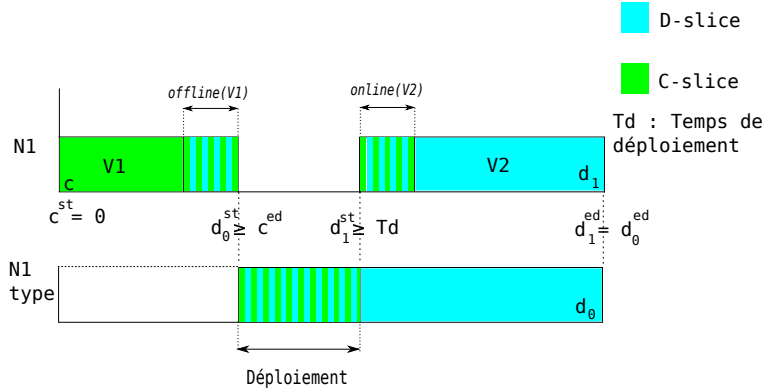


Figure 1: Exemple de configuration mettant en œuvre un changement de type; v_1 est mise hors-ligne, v_2 est allumée

Sur la figure 1, v_1 et v_2 sont deux machines virtuelles de types différents, par exemple Xen et VMWare. Pour simplifier le problème, on ne considère que des actions de démarrage et d'arrêtage pour les VMs. En effet, on pourrait remplacer celles-ci par des migrations par exemple, ce qui nécessiterait de mettre en ligne d'autres nœuds, donc d'augmenter la complexité de la configuration.

L'opération de déploiement sur le nœud n_1 se résume à:

1. mettre hors-ligne v_1 ;
2. éteindre n_1 ;
3. allumer n_1 en changeant son type, c'est-à-dire en changeant son hyperviseur.

4. démarrer v_2 ;

Le temps T_d pris par cette opération est spécifié par l'administrateur du datacenter dans la configuration de BtrPlace.

Pour que la reconfiguration puisse avoir lieu, les contraintes suivantes doivent être respectées:

- Par convention, l'opération de changement de type commence quand l'utilisation mémoire de n_1 est nulle, ie. lorsqu'aucune VM ne tourne dessus;
- $d_0^{st} \geq c^{ed}$;
- $d_1^{st} \geq T_d$;

2.2 Sans changement de type

On observe maintenant ce qu'il se passe si le type reste constant, de façon à s'assurer que l'ajout d'une nouvelle dimension ne soit pas problématique:

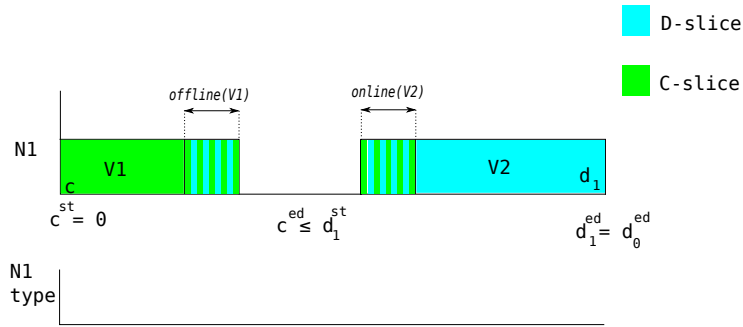


Figure 2: Exemple similaire à celui de la figure 1, mais sans changement de type, les deux VMs sont donc du même type; v_1 est mise hors-ligne, v_2 est allumée

3 Formalisation

Le placement est satisfait ssi chaque VM est bien placée sur un nœud de même type, ie.:

$$(\forall v \in \mathcal{V}), (\exists n \in \mathcal{N}), P(v) = n \Rightarrow T(n) = T(v)$$

Cette contrainte doit être implémentée dans BtrPlace via Choco.

Références

- [FH12] Gilles Muller Fabien Hermenier, Julia Lawall. Btrplace : A flexible consolidation manager for highly available applications. 2012.