

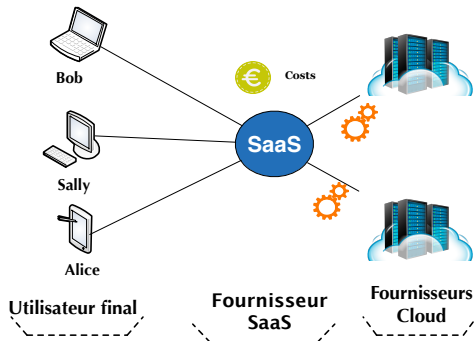
# Négociation « one-to-many » adaptative pour améliorer l'acceptabilité des services d'un fournisseur SaaS

Amro Najjar, Olivier Boissier, Gauthier Picard

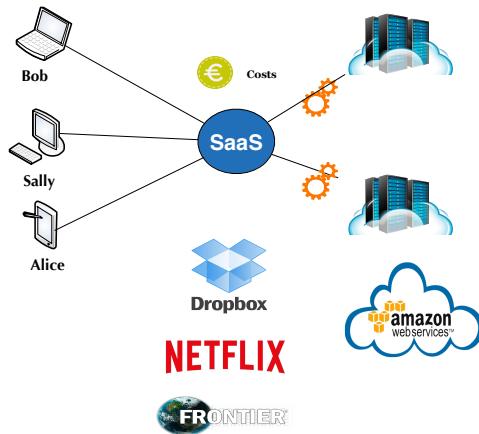
6 juillet 2017  
JFSMA 2017, Caen, France



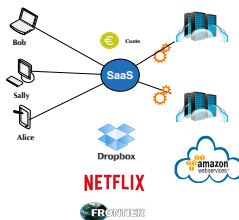
# Cloud Computing Ecosystem



# Cloud Computing Ecosystem



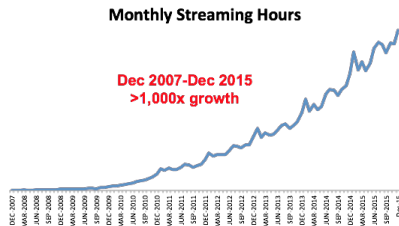
# Cloud Computing Ecosystem



## Élasticité du cloud

- ▶ Les fournisseurs SaaS peuvent louer des ressources cloud à la demande sans engagement ("pay as you go")
- ▶ Ainsi, les fournisseurs peuvent accommoder une charge variable

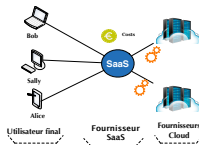
# Cloud Computing Ecosystem



## Élasticité du cloud

- ▶ Les fournisseurs SaaS peuvent louer des ressources cloud à la demande sans engagement ("pay as you go")
- ▶ Ainsi, les fournisseurs peuvent accommoder une charge variable

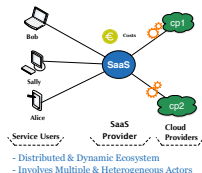
# Cloud Computing Ecosystem



## Gestion de l'élasticité [Najjar et al., 2014]

- Le fournisseur SaaS cherche à trouver un équilibre entre:
  - 1 Minimiser les coûts payés au CP
  - 2 Fournir une QoS adéquate

# Cloud Computing Ecosystem



## Gestion de l'élasticité [Najjar et al., 2014]

- Le fournisseur SaaS cherche à trouver un équilibre entre:
  - 1 Minimiser les coûts payés au CP
  - 2 Fournir une QoS adéquate

## Problème!

Dans la plupart des travaux existants, la décision de gestion de l'élasticité est prise unilatéralement par le fournisseur, les préférences et les attentes des utilisateurs finaux sont négligées

[Najjar et al., 2014]

# Proposition

Une architecture de négociation one-to-many dont le but est:

## Objectifs

- ▶ Intégrer l'évaluation subjective des utilisateurs de la qualité du service[Najjar et al., 2016a, Najjar, 2015]
- ▶ Équiper le fournisseur SaaS avec un contrôle "fine-grained" sur le taux d'acceptabilité du service tout en respectant ses contraintes budgétaires

# Proposition

Une architecture de négociation one-to-many dont le but est:

## Objectifs

- ▶ Intégrer l'évaluation subjective des utilisateurs de la qualité du service [Najjar et al., 2016a, Najjar, 2015]
- ▶ Équiper le fournisseur SaaS avec un contrôle "fine-grained" sur le taux d'acceptabilité du service tout en respectant ses contraintes budgétaires

Pour comprendre et modéliser la satisfaction des utilisateurs et les seuils d'acceptabilité subjective, nous nous appuyons sur les travaux en:

## Domaines Liés

- ① Customer expectation management [Zeithaml et al., 1993] et Psychophysique [Reichl et al., 2010]
- ② Quality of Experience (QoE) [Möller and Raake, 2014]

- 1 Contexte
- 2 Quality of Experience
- 3 Négociation « one-to-many »
- 4 Architecture de gestion de l'élasticité
- 5 Adaptive-QoE-Aware elasticity Management (AQUAMan)
- 6 Évaluation

# Qualité d'expérience

- ▶ QoE: est la qualité du service estimée **subjectivement** par l'utilisateur final (une mesure subjective et centrée utilisateur)

# Qualité d'expérience

- QoE: est la qualité du service estimée **subjectivement** par l'utilisateur final (une mesure subjective et centrée utilisateur)

## Défi

La plupart des travaux existants sont *unilatéraux* et s'appuient sur MOS (Mean Opinion Score). Cependant, MOS cache les informations sur la diversité des utilisateurs et ignore l'évolution des utilisateurs [Hobfeld et al., 2011]

# Qualité d'expérience

- ▶ QoE: est la qualité du service estimée **subjectivement** par l'utilisateur final (une mesure subjective et centrée utilisateur)

## Défi

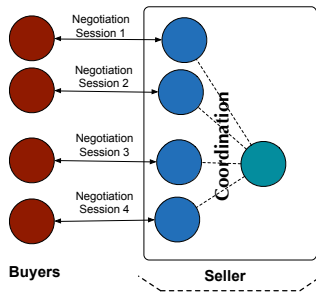
La plupart des travaux existants sont *unilatéraux* et s'appuient sur MOS (Mean Opinion Score). Cependant, MOS cache les informations sur la diversité des utilisateurs et ignore l'évolution des utilisateurs [Hobfeld et al., 2011]

## Une solution possible

- ▶ Des études empiriques récentes sur la QoE recommandent aux fournisseurs de s'appuyer sur des **percentiles** pour évaluer le taux d'acceptabilité des utilisateurs [Hobfeld et al., 2016]
- ▶ Les agents peuvent tenir compte des préférences **personnelles et évolutives** de l'utilisateur et l'impliquer dans le processus décisionnel [Najjar et al., 2016b, Najjar et al., 2016a]

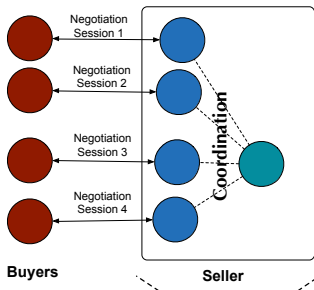
- 1 Contexte
- 2 Quality of Experience
- 3 Négociation « one-to-many »**
- 4 Architecture de gestion de l'élasticité
- 5 Adaptive-QoE-Aware elasticity Management (AQUAMan)
- 6 Évaluation

# Négociation « one-to-many »



- ▶ Permet au vendeur s de négocier simultanément avec plusieurs acheteurs
- ▶ Implique à la fois des stratégies de coordination et de négociation

# Négociation << one-to-many >>



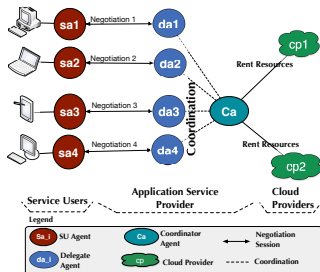
- ▶ Permet au vendeur s de négocier simultanément avec plusieurs acheteurs
- ▶ Implique à la fois des stratégies de coordination et de négociation

## Défis

- ▶ La plupart des travaux existants traitent des négociations atomiques simultanées
- ▶ Un seul accord est requis, toutes les autres sessions sont interrompues

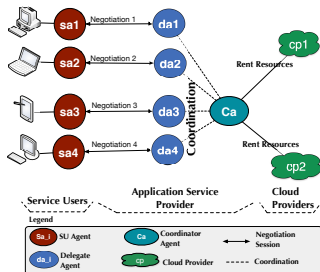
- 1 Contexte
- 2 Quality of Experience
- 3 Négociation « one-to-many »
- 4 Architecture de gestion de l'élasticité**
- 5 Adaptive-QoE-Aware elasticity Management (AQUAMan)
- 6 Évaluation

# Architecture EMan



- ▶ Intégrer l'utilisateur dans la boucle [Najjar et al., 2016a]
- ▶ Permettre une gestion de l'élasticité personnalisée et sensible à la QoE [Najjar et al., 2016b]

# Architecture EMan



- ▶ Intégrer l'utilisateur dans la boucle [Najjar et al., 2016a]
- ▶ Permettre une gestion de l'élasticité personnalisée et sensible à la QoE [Najjar et al., 2016b]

## Limitation

Le fournisseur n'a pas un contrôle précis du taux d'acceptabilité

- 1 Contexte
- 2 Quality of Experience
- 3 Négociation « one-to-many »
- 4 Architecture de gestion de l'élasticité
- 5 Adaptive-QoE-Aware elasticity Management (AQUAMan)
  - Déclenchement du mécanisme adaptatif
  - Modélisation du comportement de négociation de l'utilisateur
  - Adaptation de la négociation
- 6 Évaluation

# AQUAMan: Adaptive-QoE-Aware elasticity Management architecture

Au cours du processus de négociation, les utilisateurs peuvent accepter / rejeter le service selon leurs attentes et leur évaluation subjective de sa qualité

## Le mécanisme adaptatif d'AQUAMan

- ▶ **But:** garantir un taux d'acceptabilité prédéfini (*Goal*) tout en ne dépassant pas *RC* ou le coût moyen investi par utilisateur
- ▶ **Algorithmique:** pour ce faire, le fournisseur doit :
  - ❶ Estimer la part des utilisateurs rejetant le service
  - ❷ Construire un modèle de comportement de négociation de chaque utilisateur
  - ❸ Ajuster sa stratégie de négociation afin de restaurer le taux d'acceptabilité à ses objectifs prédéfinis tout en respectant le *RC*

# Déclenchement du mécanisme adaptatif

- ▶ Le coordinateur *ca* est notifié par chaque délégué *da<sub>i</sub>* du résultat de la session (*outcome*  $\in \{0, 1\}$ )
- ▶ *ca* exécute un algorithme d'estimation de percentile pour estimer le taux actuel d'acceptabilité du service
- ▶ Si le taux d'acceptabilité ciblé (*Goal*) n'est pas respecté, *ca* déclenche le mécanisme d'adaptation dans tous les délégués

# Déclenchement du mécanisme adaptatif

- ▶ Le coordinateur  $ca$  est notifié par chaque délégué  $da_i$  du résultat de la session ( $outcome \in \{0, 1\}$ )
- ▶  $ca$  exécute un algorithme d'estimation de percentile pour estimer le taux actuel d'acceptabilité du service
- ▶ Si le taux d'acceptabilité ciblé ( $Goal$ ) n'est pas respecté,  $ca$  déclenche le mécanisme d'adaptation dans tous les délégués

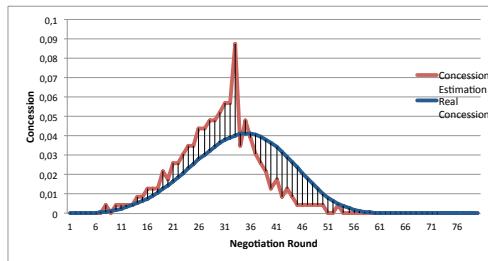
## Example

Si  $Goal = 95\%$ , le fournisseur doit vérifier que le  $(100 - 95 + 1)^{th}$  percentile égal 1. Sinon, le mécanisme adaptatif est déclenché

# Estimation de la concession de l'utilisateur

Pour apprendre le comportement de l'utilisateur,  $da_i$  surveille les concessions faites par l'utilisateur

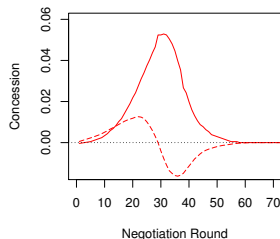
- ▶  $da_i$  estime la concession faite par un agent utilisateur  $sa_i$ . Il estime la concession faite par un agent utilisateur en comparant ses dernières offres récentes
- ▶  $da_i$  s'appuie sur sa fonction d'utilité
- ▶  $c_{sa_i}^t = M_{da_i}(o_{sa_i}^t) - M_{da_i}(o_{sa_i}^{t-1})$



# Apprendre la date limite de l'utilisateur

Les données de concession recueillies servent à établir un modèle des concessions de  $sa_i$  pour inférer sa date limite de négociation

- ▶ Dans la littérature, cela peut se faire en utilisant une régression non linéaire (c.f. [Baarslag et al., 2015])
- ▶ Cependant, contrairement à la littérature,  $da_i$  peut lancer la régression une seule fois:
- ▶  $sa_i$  utilisent une stratégie de négociation TBC
- ▶ Alors, le taux de concession de  $sa_i$  ralentit quand  $sa_i$  a déjà fait la plupart de ses concessions possibles



Les résultats de la régression sont  $r_{sa_i}^-$ , les cycles estimés avant la date limite de  $sa_i$

# Adaptation de la négociation

- ▶ *ca* choisit les séances de priorité élevée (selon leurs  $r_{sa_i}^-$ )
- ▶ La date limite de la négociation de  $da_i$  devient  $\bar{r}_i$
- ▶ *ca* ajuste la stratégie de négociation de  $da_i$  en augmentant le coût maximum possible (i.e.  $RC$ )
- ▶ Cela se fait en utilisant le mécanisme de redistribution de surplus

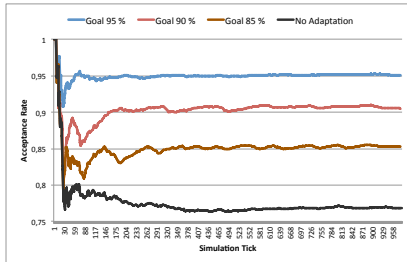
## La redistribution de surplus

- ▶ La *Tirelire* contient le *surplus<sub>i</sub>* accumulé à partir de sessions de négociation réussies
- ▶  $surplus_i = RC - Cost(\hat{o}_i)$  où  $\hat{o}_i$  est l'offre acceptée par les deux parties en session  $i$
- ▶ En utilisant ce mécanisme de redistribution excédentaire, le fournisseur assure la satisfaction de ses contraintes budgétaires

# Plan

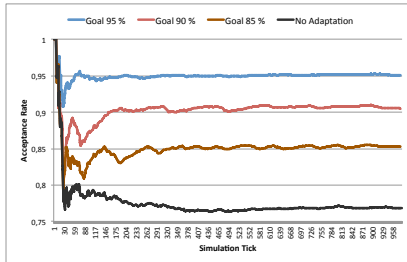
- 1 Contexte
- 2 Quality of Experience
- 3 Négociation « one-to-many »
- 4 Architecture de gestion de l'élasticité
- 5 Adaptive-QoE-Aware elasticity Management (AQUAMan)
- 6 Évaluation

# Évaluation (1): l'évaluation du mécanisme adaptatif



- ▶ AQUAMan a réussi à restaurer le taux d'acceptabilité au *Goal* prédéfini de manière précise car l'algorithme d'apprentissage a réussi à prévoir les délais de l'utilisateur (avec  $\pm 3$  cycles comme erreur moyenne)

# Évaluation (1): l'évaluation du mécanisme adaptatif



- ▶ AQUAMan a réussi à restaurer le taux d'acceptabilité au *Goal* prédéfini de manière précise car l'algorithme d'apprentissage a réussi à prévoir les délais de l'utilisateur (avec  $\pm 3$  cycles comme erreur moyenne)

## Le coût

- ▶ Plus le taux d'acceptabilité est élevé, plus le coût est élevé sans dépasser la contrainte de coût = 0.6\$
- ▶ Les coûts moyens par utilisateur étaient de 0.55\$, 0.52\$, 0.5\$ et 0.44\$ pour *Goal* = 95%, *Goal* = 90%, *Goal* = 85% et non adaptable respectivement

## Evaluation (2): la charge ( $A$ )

### Charge

| $A =$         | 20  | 40  | 80  | 160 | 320 | 640   | 1280  | 2500  |
|---------------|-----|-----|-----|-----|-----|-------|-------|-------|
| Adaptation    | 95% | 95% | 95% | 95% | 95% | 94.9% | 94.7% | 93.3% |
| No-Adaptation | 77% | 77% | 77% | 77% | 77% | 77%   | 77%   | 77%   |

- ▶ Le mécanisme s'est avéré très élastique
- ▶ Lorsque  $A \in [20, 320]$ , le taux d'acceptabilité n'est pas affecté
- ▶ Cependant, lorsque  $A \in [640, 2500]$ , le taux d'acceptabilité diminue (moins de 2 %)

## Evaluation (2): la charge (A)

### Charge

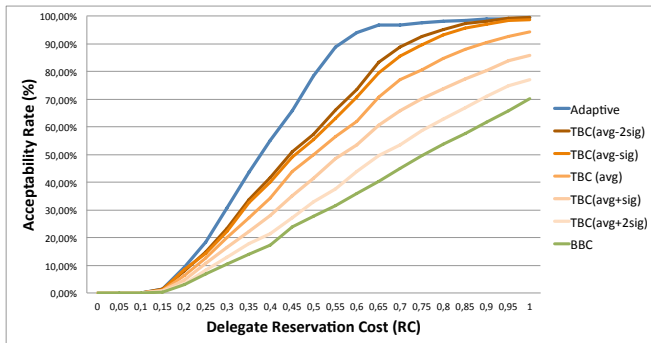
| A =           | 20  | 40  | 80  | 160 | 320 | 640   | 1280  | 2500  |
|---------------|-----|-----|-----|-----|-----|-------|-------|-------|
| Adaptation    | 95% | 95% | 95% | 95% | 95% | 94.9% | 94.7% | 93.3% |
| No-Adaptation | 77% | 77% | 77% | 77% | 77% | 77%   | 77%   | 77%   |

- ▶ Le mécanisme s'est avéré très élastique
- ▶ Lorsque  $A \in [20, 320]$ , le taux d'acceptabilité n'est pas affecté
- ▶ Cependant, lorsque  $A \in [640, 2500]$ , le taux d'acceptabilité diminue (moins de 2 %)

### Les coût de la coordination

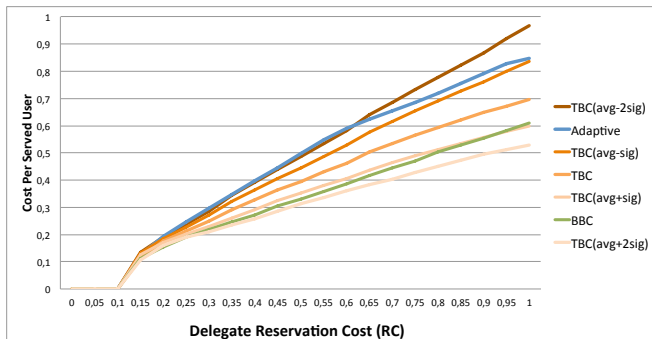
- ▶ Par rapport au mode non adaptatif, AQUAMan augmente le nombre de messages échangés entre  $ca$  et  $da_i$  par 50%
- ▶ Toutefois, vue qu'ils fonctionnent probablement sur le même data center, le coût de la communication est négligeable

# Limite du mécanisme adaptatif



- Nous avons fixé *Goal* = 100% pour pousser AQUAMan à accepter autant d'utilisateurs que possible afin d'identifier sa limite

# Coûts du mécanisme d'adaptatif



Comparaison du coût dépensé par utilisateur par AQUAMan avec d'autres stratégies non-adaptatives

# Conclusions & travaux futurs

## Conclusions

- ▶ AQUAMan équipe le fournisseur avec un contrôle "fine-grained" du taux d'acceptabilité
- ▶ Il satisfait les contraintes de coûts du fournisseur puisqu'il repose sur le mécanisme de distribution du surplus
- ▶ Il peut gérer des pics de charge qui le rend capable de gérer la nature dynamique de l'écosystème de cloud

# Conclusions & travaux futurs

## Conclusions

- ▶ AQUAMan équipe le fournisseur avec un contrôle "fine-grained" du taux d'acceptabilité
- ▶ Il satisfait les contraintes de coûts du fournisseur puisqu'il repose sur le mécanisme de distribution du surplus
- ▶ Il peut gérer des pics de charge qui le rend capable de gérer la nature dynamique de l'écosystème de cloud

## Travaux futurs

- ▶ Au lieu de l'acceptabilité binaire, le fournisseur devrait être en mesure d'assurer un niveau de satisfaction des utilisateurs prédéfini. Par exemple, 40% d'utilisateurs considèrent que le service doit être **bon** ou mieux [Hoßfeld et al., 2016], etc.
- ▶ Permettre aux agents utilisateurs d'avoir des stratégies autres que TBC

# References I



Baarslag, T., Hendriks, M. J., Hindriks, K. V., and Jonker, C. M. (2015).

Learning about the opponent in automated bilateral negotiation: a comprehensive survey of opponent modeling techniques.

*Autonomous Agents and Multi-Agent Systems*, pages 1–50.



Hobfeld, T., Schatz, R., and Egger, S. (2011).

Sos: The mos is not enough!

In *Quality of Multimedia Experience (QoMEX), 2011 Third International Workshop on*, pages 131–136. IEEE.

# References II



Hoßfeld, T., Heegaard, P. E., Varela, M., and Möller, S. (2016).

Qoe beyond the mos: an in-depth look at qoe via better metrics and their relation to mos.

*Quality and User Experience*, 1(1):2.



Möller, S. and Raake, A. (2014).

*Quality of Experience*.

Springer.



Najjar, A. (2015).

*Multi-Agent Negotiation for QoE-Aware Cloud Elasticity Management*.

PhD thesis, École nationale supérieure des mines de Saint-Étienne.

# References III

-  Najjar, A., Gravier, C., Serpaggi, X., and Boissier, O. (2016a). Modeling user expectations & satisfaction for saas applications using multi-agent negotiation.  
*In Web Intelligence (WI), 2016 IEEE/WIC/ACM International Conference on*, pages 399–406. IEEE.
-  Najjar, A., Serpaggi, X., Gravier, C., and Boissier, O. (2014). Survey of elasticity management solutions in cloud computing.  
*In Continued Rise of the Cloud*, pages 235–263. Springer.
-  Najjar, A., Serpaggi, X., Gravier, C., and Boissier, O. (2016b). Multi-agent systems for personalized qoe-management.  
*In Teletraffic Congress (ITC 28), 2016 28th International*, volume 3, pages 1–6. IEEE.

## References IV

-  Reichl, P., Egger, S., Schatz, R., and D'Alconzo, A. (2010).  
The logarithmic nature of qoe and the role of the weber-fechner law in qoe assessment.  
*In Communications (ICC), 2010 IEEE International Conference on*, pages 1–5. IEEE.
-  Zeithaml, V. A., Berry, L. L., and Parasuraman, A. (1993).  
The nature and determinants of customer expectations of service.  
*Journal of the academy of Marketing Science*, 21(1):1–12.