



Projet ANR: MOSAICO
Multi-layer Orchestration for Secured and low lAtency appliCatiOns
Projet No.: ANR-19-CE25-0012

Compte-Rendu meeting  
30-09/01-10/2020  
Troyes (UTT)

**Participants:**

- Orange : Bertrand Mathieu, Olivier Dugeon, Stéphane Tuffin
- ICD-UTT : Guillaume Doyen, Boris Kouame Ndjore, Samiha Ayed, Marius Letourneau, Hichem Magnouche, Alain Ploix
- Montimage : Edgardo Montes De Oca, Wissam Mallouli, Hoang Long Mai, Huu Nghia Nguyen
- CNRS-Loria : Thibault Cholez, Philippe Graaf

L'agenda proposé est validé.

**Accord de consortium**

L'accord de consortium a été validé et signé par tous les partenaires.  
Orange a envoyé une copie papier à chaque partenaire.

**Site Web du projet :**

Les partenaires ont remonté leurs remarques, qui ont été prises en compte par le site communication de l'UTT.  
Le site Web est maintenant OK.

**Réunions**

La prochaine réunion physique aura lieu à Nancy (Loria) les 9-10 décembre, sauf si évolution négative de la situation sanitaire.

PA Thibault : Réserver une salle et organiser la réunion.

La réunion suivante aura lieu en Mars à Paris, « hébergée » par Montimage. Montimage n'ayant pas de salle suffisamment grande, il y a 2 options : 1) on fait la réunion à Orange Gardens (salle gratuite et Montimage paie les pause-café + les



repas) ; 2) Montimage paie une salle, par ex à l'UTT qui dispose de locaux à Paris (mais payant).

PA1 Bertrand : Voir si option 1 possible

PA2 Guillaume : Si Option 1 pas possible, voir pour option 2

Nous validons le décalage des audio-conf MOSAICO au jeudi après-midi (les 2ème jeudi et 4ème jeudi du mois) de 14h à 16h

PA Bertrand : Envoyer une mise-à-jour des invitations Outlook.

### **Recrutement des thésards, stagiaires**

L'UTT a recruté ses 2 doctorants : Marius et Hichem.

Le Loria a recruté son doctorant : Philippe.

Le Loria doit recruter un ingénieur l'an prochain.

### **Dissémination**

Le projet a soumis 2 présentations aux journées Cloud (23-24 novembre à Lyon) : une de Bertrand sur le projet avec un focus sur L4S P4 et orchestrateur et une de Marius sur les framework de micro-services.

HORS réunion : La présentation de Bertrand a été acceptée, celle de Marius refusée. Bertrand réfléchit à l'organisation de la présentation.

PA Bertrand : Préparer présentations et aller la présenter aux journées.

Pour rappel, bien mentionner « This work is partially funded by the French ANR MOSAICO project, No ANR-19-CE25-0012 and supported by the french Systematic cluster. » pour tout papier et mettre les logos sur toutes les présentations.

- Présentation de Bertrand (Orange) sur algorithmes de contrôle de congestion, AQM, ECN, L4S

Bertrand a fait une présentation décrivant les principes de bases de TCP, des algorithmes de contrôles de congestion, de l'adaptation au débit de TCP, des algorithmes de type AQM (Active Queue Management), d'ECN (Explicit Congestion Notification) classique et d'ECN accurate.

En utilisation un algorithme de contrôle de congestion adapté (par ex TCP Prague), grâce à ECN Accurate, l'émetteur s'adapte plus rapidement et plus efficacement.

Accurate ECN est semblable aux infos ECN remontées avec QUIC. Il serait intéressant de voir les différences.

Actuellement, l'implémentation de L4S avec P4 se base sur un classificateur simple IP-ECN (basé sur le bit ECN) comme mentionné dans les spécifications IETF, mais il est envisagé un mécanisme plus évolué (par ex pourrait être basé sur de l'IA pour avoir une détection automatique des patterns) et ainsi éviter que des applications malicieuses se déclarent de type ECN.

A voir si cette solution peut être implémentée en P4 (mais gros doute), sinon elle serait implémentée en tant que VNF et on pourrait avoir un module léger, en relation avec le module principal, implémenté en P4 pour un traitement rapide. Cela sera raffiné dans la description des micro-services.



Cela montrerait une première découpe de L4S en micro-services et une première utilisation d'un micro-service à 2 niveaux.

PA Bertrand et Thibault : Echanger sur cette solution

- Présentation de Long sur OpenMANO pour l'orchestration de MOSAICO et déploiement d'un service

Long a présenté une démo utilisant OpenMano pour illustrer la mitigation d'attaques LDOS.

Pour cela, il déploie une VNF « Firewall » dans un environnement OpenNetVM/Flurries (Flurries permet une gestion par flow), qui va « poubelliser » les flux concernés (basés sur adresses IP) mais pas tous les flux (évite ainsi de retarder autres flux). Le module Flurries permet justement de transférer à la VNF FW uniquement les flux concernés. Cela montre la gestion de VNF par flow et non globalement.

La 1<sup>ère</sup> démo montre le cas avec une configuration statique, la 2<sup>ème</sup> avec OpenMANO, déployant, activant et configurant les VNF.

Pour cela, Long a modifié le composant Vim-Emu pour gérer le déploiement de containers OpenNetVM (qui ont des contraintes supplémentaires aux containers Docker classiques) depuis OpenMANO.

- Présentation de Long sur comparatif ONAP/OpenMANO

Long a présenté un 1<sup>er</sup> feedback de l'utilisation (+ analyse papier) de ONAP et OpenMANO.

OpenMANO n'a qu'un seul utilisateur « Admin », alors qu'ONAP a des notions de rôles/users différents.

OpenMANO n'offre pas toutes les spécifications Tosca. Il est moins complet qu'ONAP.

Il n'est pas possible de spécifier et développer un service pour OpenMANO et l'instancier tel quel dans ONAP (et vice-versa). Il faut faire des adaptations dans le fichier Tosca et le code. Cela demande donc un peu de travail, mais ce n'est pas énorme non plus.

OpenMANO est moins performant et moins complet qu'ONAP.

Montimage a fait des premiers tests sur la PF ONAP à Lannion, mais le fait de ne pas avoir accès à tout est limitant.

PA Bertrand : Voir si possible d'avoir un accès « Administrateur » ou équivalent pour Montimage à la PF ONAP.

Autre option : Montimage envisage d'installer une instance d'ONAP, mais les contraintes matérielles sont telles (capacité de traitements, nb de cœurs, etc.) que cela semble difficile (et cher).

Guillaume préfère quand même continuer sur OpenMANO, même si moins complet et abouti, car cela permet d'avoir un démonstrateur plus rapidement, c'est plus simple à utiliser et installer.



Mais pour Orange et Montimage, ONAP est une cible, donc l'idée est de commencer sur OpenMANO pour faire les premiers développements et les premières démos et on porte ensuite sur ONAP pour avoir une solution plus opérationnelle, plus « industrielle », etc.

- Présentation de Marius sur OpenNetVM (micro-services)

OpenNetVM est un framework basé sur DPDK. DPDK permet de remonter les paquets réseaux dans l'espace utilisateur en bypassant le kernel linux.

OpenNetVM fait du zero-copy en utilisant une mémoire partagée pour stocker les paquets pour faciliter le partage entre les micro-services.

Des extensions à OpenNetVM existent comme Flurries, qui fait de configuration par flux ou NFVnice, qui permet d'avoir des infos sur les traitements. Ces extensions ne sont pas concurrentes et peuvent être ensemble intégrées à OpenNetVM (à spécifier à l'installation/compilation en activant les modules souhaités). Cela pourrait être intéressant dans MOSAICO pour la gestion par flux et pour l'adaptabilité des AQM LL en fonction des traitements applicatifs (A voir si possible à prendre en compte pour la gestion des tailles des queues).

OpenNetVM permet un routage direct entre les VNF via la mémoire partagée (c'est une configuration des pointeurs sur les paquets en entrée des VNFs). Dans ce cas, il ne faudrait donc pas passer par le switch P4 pour routage local.

Actuellement le chainage est à indiquer au lancement de la VNF et la config fichier json est à lire au lancement de la VNF. Il faudrait étudier si cela peut être plus dynamique, configurable à chaud et à distance.

PA Marius : Etudier cet aspect de configuration dynamique

Les micro-services doivent être écrits en C et implémenter certaines API pour être compréhensibles pour OpenNetVM. Ce n'est donc pas n'importe quel code.

Par contre, il serait possible de chaîner des services réseaux avec des containers Dockers déployés dans l'environnement OpenNetVM. Cela serait grandement utile car cela permettrait de réutiliser des VNFs déjà développées et packagées en containers, plutôt que de tout redévelopper en C/OpenNetVM. Il faut vérifier cela :

PA Marius/Long : Vérifier utilisation de containers Docker dans chaîne OpenNetVM.

Il y a eu une discussion sur l'utilisation du Flow Table et le manager OpenNetVM. Les échanges n'ont pas aboutis à une conclusion claire. Il faut regarder cela plus en détail

PA Marius/Long : Clarifier rôle/utilisation de Flow Table et du manager dans OpenNetVM.

Enfin, selon l'architecture du nœud MOSAICO en 2 niveaux, il faudra regarder plus en détail comment remonter des flux du switch P4 vers les VNFs concernées de OpenNetVM. Ou s'il faut toujours envoyer vers une VNF de base première (qui se chargera ensuite de router, mais moins performant et pertinent).

PA Marius/Long – Bertrand/Olivier : Voir relation switch P4 et OpenNetVM

- Architecture MOSAICO



Olivier a présenté la vision d'Orange sur l'architecture réseau MOSAICO et l'architecture des nœuds MOSAICO.

L'idée est de massivement distribuer des compute nodes, pour héberger les VNFs, pour effectuer les traitements au plus près des utilisateurs et ainsi permet de faire du Low Latency. Le déploiement des VNFs serait ainsi dépendant du service final mais prendrait aussi en compte les contraintes et conditions réseaux.

Le nœud MOSAICO serait un nœud à deux niveaux : 1 switch P4 (pour les traitements réseaux bas niveaux, rapide) connecté à 1 ou plusieurs compute nodes (selon les besoins). Dans ce cas, l'objectif est de ne plus avoir de switch logiciel comme OpenvSwitch dans les compute nodes, mais que les VNFs soient directement connectés au switch P4. On rétablit ainsi les concepts, nœud réseau / nœud de traitement et chacun fait au mieux ce qu'il sait faire.

Avec le nœud à 2 niveaux, on pourra ainsi avoir un module qui fait du traitement très rapidement en P4 (mais peu complexe) et des modules plus évolués tournant dans les VNF (monitoring, sécurité, etc.) : par ex comptage en P4 + algo évolué en VNF.

Un contrôleur centralisé serait en charge de contrôler et orchestrer les nœuds MOSAICO.

Pour chaîner les services, on peut penser à un chaînage de type Segment Routing, via le switch P4 pour adresser les bonnes VNFs. A voir l'intégration possible ou pas avec un environnement OpenNetVM. En effet, La question du chaînage se pose suivant les choix architecturaux à retenir : qui indique la VNF suivante : P4 ? OpenvSwitch de compute node ? VNF directement via OpenNetVM ?

Il faudra étudier cela plus en détail et peut-être benchmarker (complexité, rapidité, efficacité, etc.)

On peut imaginer des services redondants pour basculer d'un service à l'autre pour améliorer la qualité et le routage est effectué par le switch P4.

Suite à discussion, nous avons décidé de retenir le scénario d'une architecture mono-opérateur : le switch P4 et les compute nodes sont gérés par la même entité, l'opérateur réseau.

Il faut encore réfléchir aux use-cases pertinents démontrant l'intérêt de P4 vs switch de base ou OpenFlow.

Avec P4, on ne déploie pas du code toutes les 5 minutes (nécessite un arrêt du switch, à vérifier). C'est un peu comme des mises-à-jour d'OS ou de firmware.

PA Bertrand/Olivier : Vérifier déploiement de code P4 dynamique à chaud ou à froid.

Pour rappel, il n'y a qu'un seul code P4 qui tourne à la fois sur le switch, il n'est pas possible d'en mettre plusieurs en parallèle, comme des VNFs. Par contre, il peut y avoir plusieurs tables et plusieurs actions pour faire des traitements différenciés, en fonction des configurations. C'est ainsi que nous ferons le traitement P4 souhaité.

Concernant les micro-services, on étudie la possibilité de splitter l'architecture L4S et de mettre des fonctions dans le switch P4, d'autres en VNF (par ex classification des flux ECN ou pas à base d'IA) ?



Cela demande à être raffiné par la suite, en fonction des use-cases, des besoins, des découpages en micro-services, etc.

- Services Low-Latency

Le document de base des discussions est le livrable D1.1 : <https://sharelatex.irisa.fr/project/5e2585c206d06b20e91a057e>

Il y a eu plusieurs discussions sur les critères.

Par exemple les services qui ne supportent pas la gigue : réseaux DTN, services industriels comme les SAN à distance (débit, latency courte, gigue fixe). Pour les services à gigue fixe, une solution est de passer toujours par les mêmes liens, les mêmes routeurs pour avoir toujours le même traitement.

Nous avons aussi abordé la densité de endpoints, la sécurité du service, la symétrie (ou non) des flux du services, etc.

Il faut découper la vision de latence en bouts de latence (one-way delay): delay réseau, delay processing, delay buffer, delay terminaux, etc.

Le comportement de type Bursty est important pour la gestion des files d'attente. Il faut donc voir si les services ont un tel comportement (envoi d'un seul coup) ou si c'est lissé dans le temps.

La Burstiness peut être à 2 niveaux services ou réseau. Il faut mieux expliquer.

Un rappel a été fait sur les classes de services.

Cloud Robotics : services de type capteur actionné à distance par cloud entreprise (camera, capteurs acoustiques) qui envoie des ordres aux robots

Factory Automation (5G Smart) : services de type chaîne de robot de montage (ex voiture), interco avec d'autres chaînes, etc.

Process Automation : suivi du process

Finalement, il a été décidé de ne pas focaliser sur les services de type ITS ou industriels.

Les services Cloud Robotics ou les services industriels utilisent principalement des réseaux dédiés. Il faudrait voir si ce serait possible plus tard de les fournir sur le réseau Internet avec notre architecture.

Ensuite, nous avons rempli une première version du tableau des use-cases et des critères.

Il faudra regarder des documents comme ceux de 5G ACIA ([https://www.5g-acia.org/fileadmin/5G-ACIA/Publikationen/5G-ACIA\\_White\\_Paper\\_Traffic\\_Model/WP\\_5G\\_5G\\_Traffic\\_Model\\_for\\_Industrial\\_Use\\_Cases\\_22.10.19.pdf](https://www.5g-acia.org/fileadmin/5G-ACIA/Publikationen/5G-ACIA_White_Paper_Traffic_Model/WP_5G_5G_Traffic_Model_for_Industrial_Use_Cases_22.10.19.pdf)) pour voir la classification qu'ils ont faite et comparer les valeurs des tableaux pour savoir si on peut avoir des classes proches en contraintes, en valeurs de features, etc.



Il faut reprendre un peu le tableau qui actuellement mélange critères services et critères réseaux, qui présente un peu l'architecture, etc.

PA Stéphane : Mettre à jour le tableau avec un vue orientée services et en renseignant des valeurs pour les use-cases (en indiquant les références pour justifier)

PA Tous : Voir si d'autres idées de use-cases qui pourraient compléter ceux acutels pour rentrer dans les classes de services (par ex service de travail collaboratif dentiste-prothésiste pour travailler ensemble en direct sur les plans).

