

Architecture de Référence pour le Edge Computing: Application au Système de Contrôle des Risques dans le Transport

Zakaria Benzadri*, Abdeldjalil Boultifat**, Sami Ressler Zair***
Faiza Belala****

*zakaria.benzadri@univ-constantine2.dz

**abdeldjalil.boultifat@gmail.com

***samizair95@gmail.com

****faiza.belala@univ-constantine2.dz

LIRE Laboratory, University of Constantine 2-Abdelhamid Mehri Constantine, Algeria.

Résumé. Le Edge Computing est un nouveau paradigme informatique qui est désormais le plus attirant dans le milieu industriel aussi bien dans le milieu académique. Il est actuellement considéré comme une évolution du Cloud Computing grâce à sa capacité de traiter et stocker les données d'une manière décentralisée au niveau proche de leurs sources et non plus sur l'infrastructure Cloud. Le Edge Computing présente un moyen très efficace pour une meilleure fiabilité, un faible temps de latence et une bonne réduction des besoins en bande passante dans les centres de données. Néanmoins, et en dépit de ces points forts, le Edge Computing rencontre des difficultés à cause de l'hétérogénéité et la complexité des différentes couches constituantes (Cloud, IoT, Mist, etc.). Ceci engendre une définition indéterminée des éléments impliqués et offre une mauvaise représentation de l'architecture Edge. Dans ce papier, nous mettons en œuvre une architecture de référence pour le Edge Computing; d'un haut niveau d'expressivité supportant l'ensemble de ses concepts de base. Ensuite, nous concrétisons les résultats obtenus, via l'application de l'étude de cas d'un système de contrôle des risques dans le transport, basé Edge Computing.

1 Introduction

L'année 1961 a vu l'apparition de la notion d' "Utility Computing" –introduite par John McCarthy Foster et al. (2008). Cette vision futuriste consiste à vendre la puissance de calcul et même les applications spécifiques comme un service public. A l'époque, l'idée novatrice de la consommation des services informatiques n'a pas vu le jour, car, les technologies matérielles et logicielles n'étaient tout simplement pas prêtes. De nos jours, et grâce aux avancées des technologies de l'information et de la communication, cette idée a incarné à nouveau, elle est apparue sous une autre forme. Le Cloud Computing met en œuvre l'idée d' "Utility Computing" en offrant une allocation dynamique des ressources informatiques selon trois modèles de prestations de service : l'infrastructure en tant que service (IaaS), la plateforme

en tant que service (PaaS), et l'application en tant que service (SaaS). D'un point de vue déploiement, le Cloud Computing propose aussi quatre modèles : l'infrastructure est exploitée pour le compte d'une entreprise –Cloud privé; l'infrastructure est partagée entre plusieurs entreprises –Cloud communautaire; l'infrastructure est mise à la disposition du grand public –Cloud public; et enfin, l'infrastructure est une combinaison de deux ou plusieurs des modèles de déploiement précédents –Cloud hybride. En effet, le Cloud Computing vise à rendre les ordinateurs invisibles Michael et al. (2010), en offrant de grandes capacités de calcul et de stockage à la demande Zhang et al. (2015). Cependant, et avec l'apparition de l'Internet des objets (IoT) et son impacte sur la vie quotidienne des utilisateurs, l'aspect centralisé du Cloud Computing n'est plus adapté aux besoins de cette dernière, à cause de la distance éloignée ainsi que le nombre croissant des utilisateurs et des objets connectés; selon l' "International Data Corporation" le nombre des périphériques connectés va atteindre les 29 milliards d'ici 2020 Buyya et al. (2009).

Le Edge Computing est un nouveau paradigme de l'informatique utilitaire, qui se voit plus adapté pour l'Internet des objets grâce à sa capacité de traiter et stocker les données collectées des capteurs au niveau des extrémités logiques du réseau. Ce paradigme consiste à apporter des solutions alternatives aux différents problèmes posés par le Cloud Computing Chang et al. (2019), ayant pour but de combler ses insuffisances et d'offrir une meilleur rapidité et efficacité. Le Edge Computing, comme son nom l'indique, réalise les calculs sur des micro serveurs situés en bordure du réseau, et non plus sur des infrastructures centralisées Jararweh et al. (2014). Cette approche vise donc à décentraliser le traitement des données afin de garantir une latence réduite, un coût minimal et une bande passante maîtrisée Cao et al. (2018). En dépit de ces points forts, le Edge Computing présente plusieurs défis qui freinent sa croissance et son adoption. Nous considérons dans ce travail les défis suivants Varghese et al. (2016) :

D1 : Partitionnement de la charge de travail. Le Edge Computing est constitué de plusieurs couches chacune à une capacité de calcul différente, ce qui engendre un problème de partitionnement de la charge de travail sur ces différentes couches afin d'optimiser le traitement des données.

D2 : Mise à l'échelle rapide. La mise à l'échelle rapide des systèmes Edge constitue un autre défi majeur en réponse au nombre croissant des objets connectés (IoT).

D3 : Diversité des dispositifs. Le Edge Computing fait face à un problème d'identification et d'observation globale des différents dispositifs connectés à cause de leurs nombre ainsi que la diversité de leurs données générées.

En réalité, ces défis engendre une définition indéterminée des éléments impliqués dans un système Edge Computing et conduit a une mauvaise représentation de son architecture. Rajoutant à cela, les études menées sur la modélisation des architectures Edge Computing montrent que la plupart de ces approches sont ad-hoc, complexes, difficiles à adopter et à réutiliser. En effet, les différentes approches proposées sont principalement des approches à intérêt industriel (ou technique). Ces approches offrent des solutions spécifiques pour traiter un problème particulier par rapport à une technologie imposée par un acteur de l'industrie, ou bien un besoin signalé par un utilisateur.

Pour surmonter ces défis (*D1*, *D2* et *D3*), la modélisation de l'architecture Edge représente une meilleure solution. Par ailleurs, et à défaut de pouvoir réduire la complexité des systèmes

Edge en général, il a fallu la maîtriser en donnant une illusion de simplicité. Cette illusion est possible grâce à l'abstraction, qui permet de réduire la quantité d'informations. D'une part, en mettant en avant les caractéristiques essentielles du système selon toutes ses facettes. D'autre part, en ignorant ses caractéristiques secondaires. Les architectures de références forment le moyen, le plus approprié, pour une telle description du système ainsi que son comportement. Pour cela, nous définissons une architecture de référence, nommée "*EECA -Enhanced Edge Computing Architecture*", fortement expressive et offre une meilleure séparation entre les différentes couches identifiées, à savoir : *Dispositif Edge, Mist Computing, Edge Computing et Cloud Computing*. Ensuite, nous illustrons et validons cette architecture à travers la modélisation et l'instanciation d'une étude de cas du système de contrôle des risques dans le transport "*Edge Risk Control System*".

Le papier est principalement structuré en trois parties à savoir :

- **La première partie**, est consacrée à l'établissement d'une synthèse récapitulative et une étude approfondie de l'état de l'art et les travaux existants autour de la modélisation des architectures Edge Computing.
- **La deuxième partie**, représente la partie principale de ce travail dont la sémantique de l'architecture proposée ("*Enhanced Edge Computing Architecture*") est décrite.
- **La troisième partie**, comprend la validation de l'approche proposée à travers une étude de cas concrète en relation avec les systèmes de contrôle des risques dans le transport.

2 Edge Computing

De nombreux auteurs ont tenté de définir le Edge Computing selon différents points de vues. En guise de synthèse des définitions largement utilisées dans l'industrie et le domaine académique (Cao et al. (2018), Yu et al. (2018b) et Trifiro et Smith (2018b)), nous introduisons une définition intégrée qui offre une vue plus spécifique et objective du Edge Computing :

"Le Edge Computing est un nouveau paradigme informatique qui représente une architecture distribuée et ouverte. Il consiste à apporter les capacités de traitement et de stockage aux extrémités logiques du réseau et sans avoir le besoin de transférer les données vers les centres de données. Ceci, en réponse aux contraintes de latence et de bande passante largement rencontrées dans le Cloud Computing".

2.1 Principe de Fonctionnement

Pour une vision plus éclaircie du Edge Computing et son principe de fonctionnement, la figure 1 illustre ses couches de base. En particulier, la figure décrit une couche intermédiaire appelée "Near-end" entre les deux couches "Far-end" et "Front-end". Cette couche représente une couche supplémentaire dotée des capacités de traitement qui permettent au Edge Computing d'être plus efficace via le rapprochement du traitement des données issues des différents dispositifs de l'IoT et les micro centres de données.

Le Edge Computing et le Fog Computing sont deux nouveaux concepts considérés comme le future de l'informatique grâce aux solutions qu'ils apportent pour surmonter les défis ren-

Architecture de Référence pour le Edge Computing

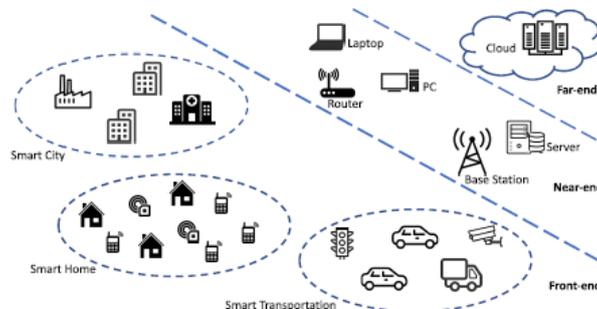


FIG. 1 – Principe de fonctionnement du Edge Computing Yu et al. (2018a)

contrés par le Cloud Computing. Ces deux paradigmes étendent le Cloud Computing pour que les différents ressources de stockage et traitement des données peuvent être déployées séparément d'une manière optimale. Cela, dans le but d'améliorer les performances et de se rapprocher des utilisateurs. En effet, Le Edge et le Fog Computing permettent le traitement des données à proximité de leurs source, mais de manière différente : "le Fog localise l'intelligence au niveau du réseau local, alors que le Edge localise une partie de son intelligence au niveau de la source des données collectées par les capteurs".

2.2 Éléments Architecturaux

Parmi les éléments de base du Edge Computing, nous nous intéressons plus particulièrement aux : "Edge Node", "Latency", "Software Edge", "Edge Micro Data Center", "Edge Devices" Trifiro et Smith (2018a), "Edge Infrastructure", "Data Reduction" CB Insights (2019), "Edge Server" GE Digital, Driving Digital Transformation et "Mist Computing". Le tableau suivant 1 décrit les concepts de base cités précédemment.

2.3 Caractéristiques

La particularité du Edge Computing se manifeste dans ses trois caractéristiques (voir figure 2) : Connectivité, Calcul, Contrôle et Sécurité Bradicich (2018). Nous les détaillons dans ce qui suit.

Connectivité ("Connectivity"). Le Edge Computing facilite l'échange des informations entre les différents éléments connectés au réseau, tels que les dispositifs, les individus et les objets. Il permet de minimiser la latence étant donné que de nombreuses applications dans le Edge nécessitent un aperçu et un contrôle en temps réel. De plus, il permet de réduire la bande passante et d'améliorer la fiabilité, cela, en conservant au niveau Edge une copie des données envoyées vers le Cloud.

Calcul ("Compute"). Le Edge Computing traite l'ensemble des informations collectées, offre un accès aux applications et fournit des informations supplémentaires sur ses différents éléments et son environnement. Il permet aussi d'éviter la duplication, en se

Concept	Description
"Edge Node"	représente un nœud faisant partie d'une architecture Edge qui réunit un ensemble de ressources informatiques, exploité dans le calcul et le traitement des données.
"Latency"	c'est le temps nécessaire de transmission des données sur le réseau entre deux ou plusieurs extrémités.
"Software Edge"	représente l'ensemble des applications logicielles déployées à des points proches à l'utilisateur final au sein de l'infrastructure Edge.
"Edge Infrastructure"	représente un ensemble distribué de centres de données qui offrent des ressources avec une forte densité entre les périphériques ("Edge devices") et le Cloud.
"Edge Micro Data Center" (EMDC)	représente un centre de données qui applique le concept de centre de données modulaire à petite échelle. Il est conçu pour résoudre les problèmes liés aux différents types de calcul, y compris ceux liés aux ressources disponibles dans une zone donnée.
"Edge Devices"	représente l'ensemble de périphériques et dispositifs tels que les smartphones, les ordinateurs portables et les capteurs qui ont pour rôle de générer et collecter des données.
"Edge Server"	peut être décrit comme un ensemble d'ordinateurs dédié à l'exécution des applications aux extrémités du réseau.
"Data Reduction"	consiste à utiliser un point intermédiaire pour réduire le volume des données et les transmettre d'une manière intelligente entre le producteur et le consommateur.
"Mist Computing"	permet de fournir un ensemble de services et dispositifs pour améliorer l'évolutivité, la reconfigurabilité, et la communication entre les différentes applications IoT (Preden et al. (2015) , Liyanage et al. (2016)), en se basant sur l'environnement physique et la situation du réseau local.

TAB. 1 – *Concepts de base sur l'Edge Computing*

basant sur la notion de filtrage des données et en vérifiant leurs existence dans le Cloud ; si les données à envoyer sont déjà stockées, ces derniers ne seront pas envoyées.

Contrôle et Sécurité ("Control and Security"). Le Edge computing fournit des informations qui peuvent être utilisé pour contrôler et prendre des décisions sur les dispositifs connectés. Il permet aussi de maintenir la conformité des transferts de données à distance, en satisfaisant les exigences et les politiques des différentes entreprises. Par exemple, certains pays obligent des politiques de confidentialité pour réduire les menaces et protéger ainsi leurs citoyens contre l'usurpation d'identité et interdisent aux entreprises de transférer les données personnelles de leurs citoyens hors de leurs frontières.

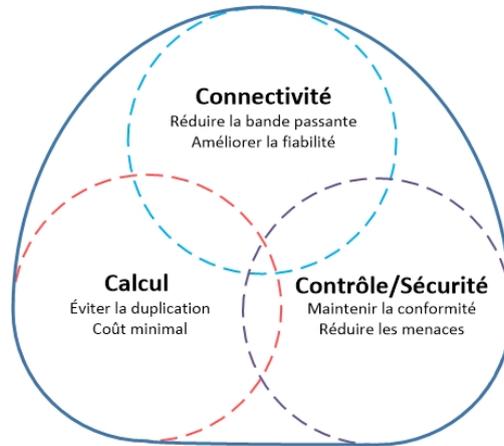


FIG. 2 – Caractéristiques du Edge Computing

3 Approches de Modélisation

Après une étude approfondie des différents travaux de modélisation existants dans la littérature, nous avons constaté qu'aux premiers stades de la recherche sur le Edge Computing, il y avait seulement certaines tentatives à caractère technique, c'est au cours de ces dernières années, que quelques approches à caractère fonctionnel ont émergé pour la modélisation de ce type de systèmes. Nous identifions dans nos diverses lectures deux styles architecturaux fortement adoptés dans ce contexte, à savoir, les architectures multi-couches pour les travaux à caractère technique et les architectures orientées services pour les travaux à caractère fonctionnel. Partant de ce constat, nous classifions les approches de modélisation des systèmes Edge en deux grandes catégories ; selon le contexte considéré et les aspects supportés :

- *Travaux à caractère Technique*, traitant les aspects technologiques du Edge Computing ;
- *Travaux à caractère Fonctionnel*, spécifiant les aspects fondamentaux des systèmes Edge.

3.1 Travaux à caractère Technique

Les travaux relatifs à cette catégorie concernent principalement la modélisation des aspects purement techniques du Edge Computing. L'aspect technique d'une architecture représente l'organisation logique des différents éléments selon des standards de construction, ainsi que les relations entre ces différents éléments Kurbel (2008).

Les auteurs de l'architecture présentée dans Li et al. (2018) proposent une architecture en cinq couches. Chaque couche fournit un service à la couche adjacente (voir la figure 3). La première couche représente le noyau de cette architecture, le Cloud centralisé au sein de cette couche est représenté par les Méga-centres de données. La deuxième couche, tels que la couche précédente mais avec la particularité de fournir une certaine distribution et des centres

de données moins puissants. Ces deux couches représente le Cloud Computing. Ensuite vient le Edge Computing qui est représenté dans cette architecture via les trois couches suivantes ; La troisième couche contenant des centres de données modulaire et des conteneurs. La quatrième couche possédant des routeurs, des passerelles et des points d'accès. Et enfin, la cinquième couche, qui représente l'ensemble des capteurs et des différents dispositifs tels que les caméras, les smart-phones qui sont des terminaux du réseau. Notant seulement que la distance entre l'utilisateur et les différentes couches de cette architecture n'est pas la même ; tout en s'éloignant du noyau la distance et la capacité diminue.

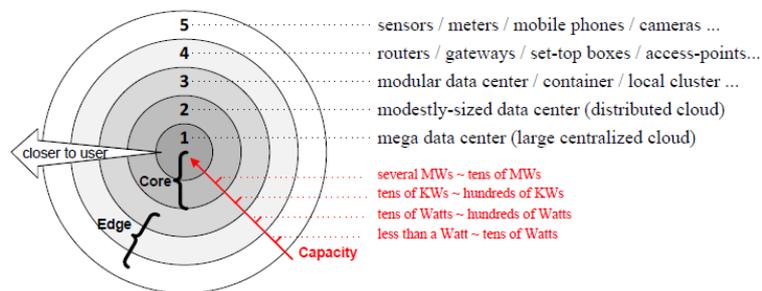


FIG. 3 – Architecture multi-couches du Edge Computing Li et al. (2018)

Dans le travail de Sehout et al. (2018), les auteurs proposent une architecture "ArchMlt2" composée de trois couches, une couche pour les dispositifs IoT, une couche Edge et une couche Cloud. La couche "IoT devices" contenant quatre types d'éléments architecturaux qui sont : les capteurs, les contrôleurs, les dispositifs hybride et les utilisateurs. La "Couche Edge" illustre la plateforme Edge Computing, équipée par les Serveurs Edge (Serveur de Sécurité, Serveur de Données, Serveur d'Application) dans le but de traiter et gérer les données issues de la couche précédente. Cette couche se connecte à la couche inférieure à travers les "Gateway Edge Routeurs". Enfin, la "Couche Cloud", représente la couche supérieure de l'architecture qui permet d'assurer le stockage et le traitement des données arrivant de la couche inférieure via des protocoles de communication.

3.2 Travaux à caractère Fonctionnel

Très peu de travaux concerne principalement des aspects fonctionnels relatifs à la gestion des ressources dans le Edge, les processus d'affaires et la composition des services.

Les auteurs de Intel SAP (2018), ont proposé une architecture orientée service "SAP Leonardo IOT Edge" qui contrôle les grandes quantités de données générées par les dispositifs IoT et plus précisément les dispositifs industriels au niveau de leurs source. Cela, en proposant des services qui permettent de stocker les données au niveau du nœud Edge au lieu de les transmettre au SAP Cloud Platform (voir la figure 4). Les services de persistance ("Persistence services") permet le stockage local des données. Les service d'analyse en continu ("Streaming analytics") consiste à analyser les flux de données provenant des différents dispositifs de l'IoT afin de trouver des problèmes ou des exceptions, et de les signaler. Les services ("Business-

Architecture de Référence pour le Edge Computing

essential functions") assure la continuité de fonctionnement des différents dispositifs de l'IoT dans le cas où il y a une coupure de la connexion avec SAP Cloud Plateforme, en fournissant des processus métiers de secours. Les services de "Predictive modeling") offre des modèles prédictifs pour analyser les données disponibles, ces modèles agissent au niveau proche des données pour améliorer les prédictions. Enfin, les services "Machine learning" permet d'appliquer les algorithmes avancés de l'apprentissage automatique pour l'analyse des données.

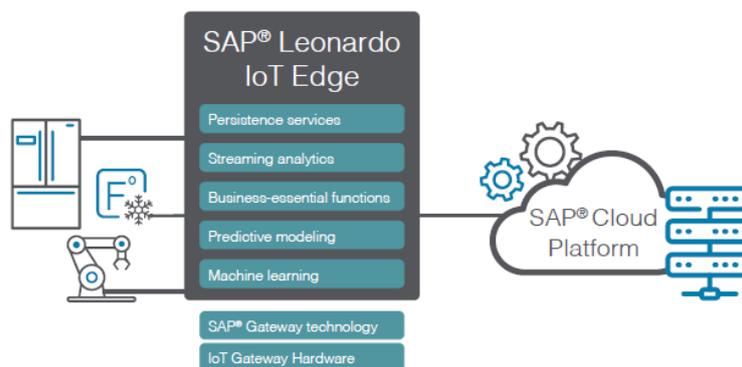


FIG. 4 – Architecture SAP Leonardo IOT Edge Intel SAP (2018)

Dans la même direction, les auteurs de Villari et al. (2016) ont proposé l'architecture orientée service nommée : "Osmotic Edge Computing". Cette architecture est inspirée d'un processus chimique appelé l'osmose qui permet aux molécules du solvant de passer d'une solution diluée à une autre plus concentrée à travers une membrane. En suivant ce même principe de fonctionnement, l'architecture proposée montre comment les micro-services peuvent passer du Cloud vers le Edge ou vice versa, dynamiquement via une membrane semi-perméable. Et cela dans le but de répondre aux exigences des différentes applications qui nécessitent des capacités du Cloud et du Edge à la fois, pour assurer les critères de la qualité de service (QoS) tels que la disponibilité, le coût et le débit. Ces micro-services passent du Cloud vers le Edge afin de satisfaire les besoins de faible niveau et les besoins de haut niveau et aussi afin d'assurer la disponibilité des ressources au niveau du "Edge micro data centers" et le "Cloud data centers" (voir la figure 5).

3.3 Comparaison des Travaux

Nous procédons dans cette section à la comparaison des travaux (voir tableau des travaux 3) présentés dans les sous-sections précédentes, pour se faire, nous nous concentrons sur le degré d'expressivité des architectures en termes de leurs fonctionnement, leurs éléments et la distribution de leurs différentes couches (voir tableau des critères d'expressivité 2); jugés les plus pertinents par rapport à notre étude.

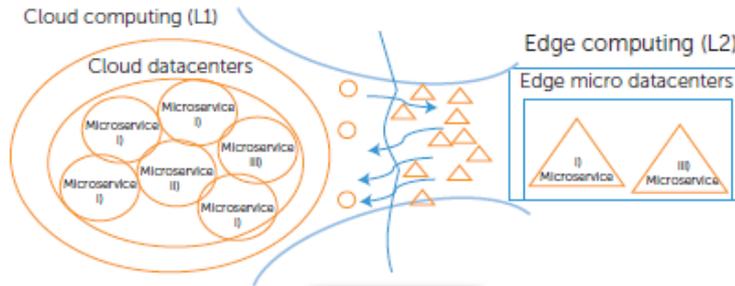


FIG. 5 – Architecture Osmotic Edge Computing Villari et al. (2016)

CP	Capteurs
CT	Contrôleurs
GW	Gateways
DC	Micro data Center
PC	Protocole de Communication
DT	Données
SR	Services
CL	Couche Cloud
EL	Couche Edge
DL	Couche Edge Devices
ML	Couche Mist

TAB. 2 – Aspect d'expressivité

D'après l'étude approfondie des différents travaux existants –dont les résultats de comparaison sont présentés dans le tableau précédent, nous avons dégagé les constatations suivantes :

- Les travaux à caractère Technique proposent des solutions *spécifiques* pour traiter un problème d'ordre technologiques ; souvent imposé par un fournisseur ou un utilisateur. Dans cette lignée, ces travaux sont principalement basés sur le style architecturale multi-couches et concernent que des questions relatives à la gestion des ressources et de l'interopérabilité entre les différents fournisseurs dans le Edge.
- Les travaux à caractère Fonctionnel proposent des solutions *abstraites mais non-génériques* ; le style architecturale orienté service utilisé n'est pas capables d'exprimer autant d'aspects théoriques dans les architectures Edge. Dans cette catégorie, les travaux se montrent plus appropriés pour assurer l'intégrité de la découverte et la composition des services dans le Edge.
- D'une manière générale, et en terme d'expressivité, nous constatons, en particulier, que la spécification des aspects relatifs aux données (DT), à la couche Mist (ML) et aux protocoles de communication (PC) ont été occultés par rapport aux autres aspects dans le Edge. D'autre part, nous remarquons aussi que très peu de concentration a été orientée vers les aspects de micro centres de données (DC), contrairement aux aspects liés au Cloud (CL).

	Élément				Mécanisme de fonctionnement			Couche			
	CP	CT	GW	DC	PC	DT	SR	CL	EL	DL	ML
Architecture proposée dans Li et al. (2018)	+	+	+	+	-	-	-	+	+	-	-
Architecture proposée dans Sehout et al. (2018)	+	+	+	-	+	-	-	+	+	+	
Architecture proposée dans Intel SAP (2018)	+	+	+	-	-	-	+	+	+	+	-
Architecture proposée dans Villari et al. (2016)	-	-	+	+	-	-	+	+	+	-	-

TAB. 3 – Comparaison des architectures Edge Computing

4 Proposition de l'Architecture de Référence "Enhanced Edge Computing Architecture"

La croissance des dispositifs connectés, due principalement à l'usage de l'Internet des objets à large échelle, a introduit de nouveaux défis pour la conception d'architectures distribuées Edge Computing. Cette dernière est devenu incontournable pour gérer l'ensemble de ces dispositifs connectés. En effet, les difficultés se manifeste principalement dans l'organisation logique des différents composants et leurs déploiement physique. Effectivement, les architectures des systèmes distribuées sont articulées autour de deux grands axes : (1) des composants et (2) des connecteurs assurant l'interaction entre ces composants afin d'assurer le partage des ressources Sugathadasa (2017). La modélisation et la construction d'une architecture distribuée est fortement motivées par le besoin d'appliquer ces solutions sur différents domaines afin d'atteindre des objectifs spécifiques et opérationnels, tels que le domaine du transport et le domaine médical Sugathadasa (2017).

Dans cette section nous allons décrire notre contribution concernant la proposition d'une architecture Edge Computing, prenant en charge les concepts du Edge Computing et ses aspects structurels. Ensuite, nous détaillons chaque couche identifiée et le rôle de ses constituants.

4.1 Principe de notre approche

L'architecture Edge s'appuie généralement sur le traitement et le stockage des données au niveau proche de leurs sources. La figure 6 illustre la distribution globale pour le cas de l'architecture de référence proposé. Afin de gérer la complexité de l'architecture Edge Computing, nous avons eu recours à l'un des principaux concepts du génie logiciel, la séparation des préoccupations De Win et al. (2002). Ce principe consiste à considérer séparément les différents aspects d'un problème, afin d'en maîtriser la complexité. En effet, la séparation des préoccupations fournit un support méthodologique permettant de décomposer un système en un ensemble d'éléments compréhensibles, chacun s'intéressant à une seule préoccupation. Ainsi, nous adoptons un style architectural en couches pour modéliser l'architecture Edge Computing. En particulier, nous identifions quatre couches : dispositifs Edge ("Edge Devices"), Mist, Edge et Cloud (voir la figure 7). Chaque couche possède ses propres responsabilités et s'appuie sur les autres couches pour assurer ses tâches.

La couche dispositifs Edge ("Edge Devices Layer") contient : les Capteurs de Haute Qualité ("High Quality Sensors"), Capteurs de Temps Critique ("Time Critical Sensors"), Capteurs de Haute Vitesse ("High Velocity Sensors"), Dispositifs Portable et Capteurs ("Wearables

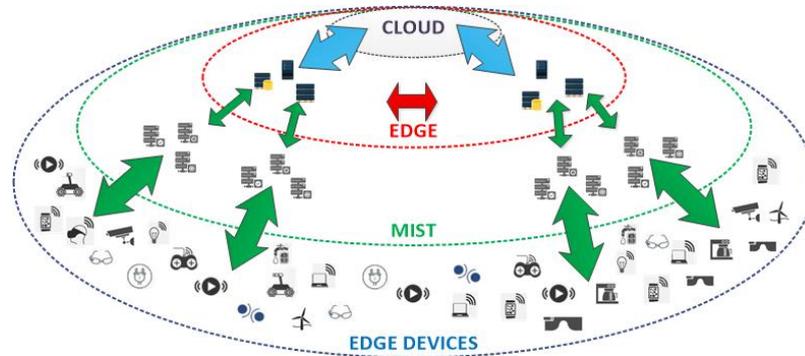


FIG. 6 – Vue globale de l'architecture proposée

And Sensors") et les Contrôleurs ("Controllers"). Alors que la couche Mist ("Mist Layer") comprend : un "Mist Request Broker", "Dispatch Unit", "Analysis Unit" et "Control Unit". De plus, la couche Edge ("Edge Layer") contient : "Edge Gateway Router", "Micro Data Center", "Application Server", "Cloud Gateway Server". Enfin, la couche Cloud ("Cloud Layer") représente un espace de stockage de données. La projection des différents éléments selon la vue globale de l'architecture (de la figure 6) est schématisée dans la figure 7.

Dans les sous-sections qui suit, nous allons détailler les différentes couches de notre architecture de référence "Enhanced Edge Computing" (EECA) et la préoccupation de chaque élément ainsi que leurs dépendances.

4.2 Couche Dispositifs Edge ("Edge Devices Layer")

Le premier niveau de notre architecture représente l'ensemble des dispositifs. Ces dispositifs sont décomposés en cinq catégories de composants ; cela en se basant sur la façon que leurs données seront traitées (voir la figure 7 .a). Le tableau (4) décrit l'ensemble des composants appartenant à cette couche.

4.3 Couche Mist ("Mist Layer")

Le deuxième niveau de notre architecture permet de regrouper les différents dispositifs du premier niveau. Cela en donnant la possibilité de communiquer les ressources disponibles entre eux. Rajoutant à cela, ce niveau permet d'effectuer des calculs sur les données en temps réel. Cette couche communique avec le troisième niveau (couche Edge) à travers une couche intermédiaire constituée des moyens de communication comme le WiFi et facilite la gestion du premier niveau pour ce dernier (voir la figure 7 .b). Dans le tableau (5) nous allons définir chaque élément appartenant à cette couche.

Architecture de Référence pour le Edge Computing

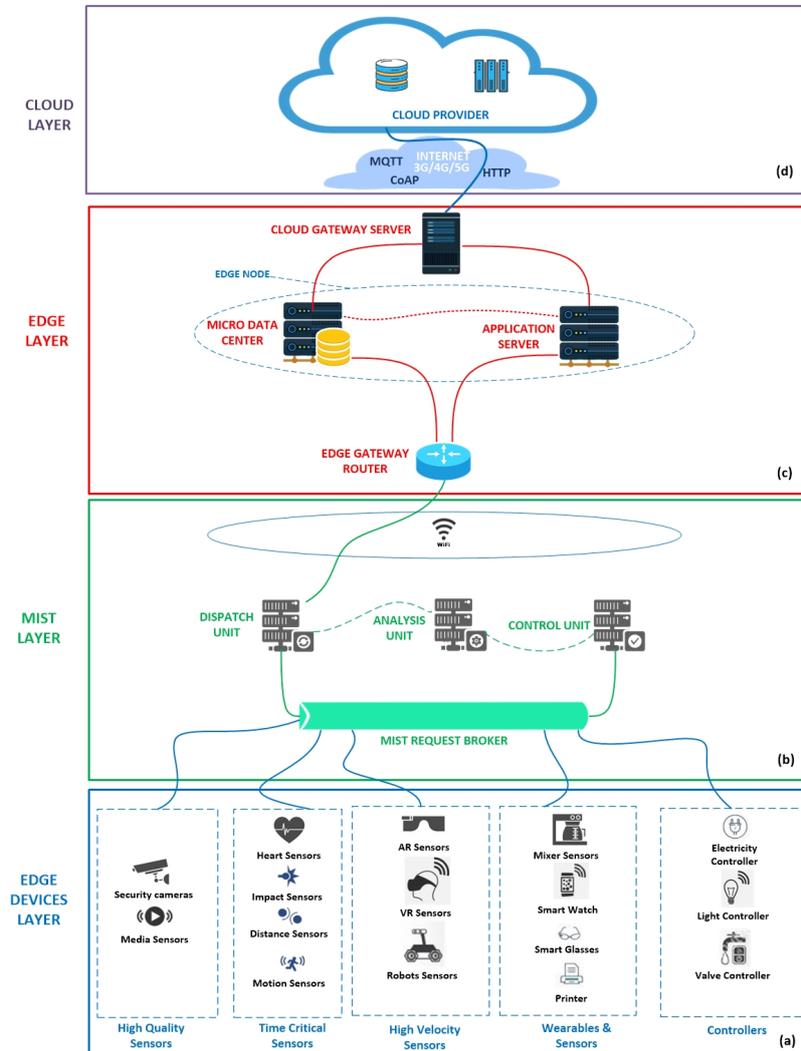


FIG. 7 – Architecture de référence "Enhanced Edge Computing"

4.4 Couche Edge ("Edge Layer")

Le troisième niveau de notre architecture permet le traitement et l'analyse des données provenant du deuxième niveau (couche Mist) en temps réel, et cela grâce à sa proximité de la source des données. De plus, il a une capacité considérable qui lui permet de stocker les données sans avoir le besoin de les exposer au quatrième niveau (voir la figure 7 .c). Dans le tableau (6) nous allons définir chaque élément appartenant à cette couche.

Groupe	Description
Capteurs de Haute Qualité ("High Quality Sensors")	Englobe les capteurs dont leurs données générées ne permettant aucune perte de qualité à cause d'un traitement intermédiaire avant d'arriver à leurs destination finale.
Capteurs de Temps Critique ("Time Critical Sensors")	Englobe les capteurs où le traitement de leurs données doit s'effectuer en temps réel, chaque retard dans le traitement engendre un échec ou un changement dans le comportement du système qui dépend sur ces capteurs.
Capteurs de Haute Vitesse ("High Velocity Sensors")	Englobe les capteurs où leurs données générées vont être compressées d'une façon intelligente sans aucune perte d'information au niveau d'un point intermédiaire ("Edge node") avant d'arriver à leur destination finale. Cela permet aussi d'éviter la duplication des données.
Dispositifs Portable et Capteurs ("Wearables And Sensors")	Englobe les capteurs où leurs données ne nécessitent pas une préservation de la qualité et n'exigent pas une haute priorité de traitement.
Contrôleurs ("Controllers")	Englobe les dispositifs qui consomme les données afin d'interagir avec l'environnement où ils sont situés.

TAB. 4 – *Décomposition des dispositifs de la couche Edge Devices*

4.5 Couche Cloud ("Cloud Layer")

Le quatrième niveau de notre architecture permet le stockage des données provenant du troisième niveau ("Edge Layer") à travers une couche intermédiaire constituée d'un ensemble de protocoles de communication tels que MQTT, HTTP, COaP (voir la figure 7 .d). Cette couche représente une abstraction de la partie Cloud.

5 Application de l'étude de cas

Dans cette section, nous proposons une solution basée sur l'architecture de référence "EECA" proposée, dans le but de surmonter les problèmes rencontrés liées aux systèmes de contrôle des risques pour le transport.

5.1 Motivation

Depuis très longtemps, les moyens de transport ont joué un rôle prépondérant, en contribuant de manière positive et efficace à l'amélioration de la qualité de vie de l'être humain tels que le déplacement des individus et des marchandises. Ainsi, de différents modes de transport ont apparus, tels que les modes terrestre, maritimes, fluviaux et aériens. De nos jours, le mode de transport le plus fréquemment utilisé est le mode terrestre qui comporte le transport routier. Ce dernier rassemble les véhicules et les bus. En outre, le transport ferroviaire qui rassemble les trains a aussi donné un saut dans le développement économiques et commerciales des pays, ce qui a engendré une forte augmentation de leurs usages Kahlouche et Chaib (2017). Par conséquent, de nouvelles inventions dans le mode de transport ferroviaire tels que le "tram",

Architecture de Référence pour le Edge Computing

Élément	Rôle	Code	Représentation Graphique
"Mist Request Broker"	Permet d'assurer le routage des données des différents dispositifs vers les unités de traitement et le routage des résultats vers leurs destinations. Il permet aussi de désactiver les dispositifs inactifs et les réactiver lorsqu'ils sont sollicités.	M0001	
"Dispatch Unit"	Assure la redirection des données provenant du premier niveau selon leurs types. Si ces derniers sont de type "temps critique" et dans le contexte, elles seront dirigées vers l'unité d'analyse ("Analysis Unit"). Sinon les données vont être dirigées directement vers le troisième niveau.	M0002	
"Analysis Unit"	Permet d'analyser et traiter les différentes données arrivant de l'unité de redirection ("Dispatch Unit") afin de prendre des décisions et transmettre les résultats vers l'unité de contrôle ("Control Unit").	M0003	
"Control Unit"	Récupère les résultats provenant de l'unité d'analyse ("Analysis Unit") et les envoie vers les dispositifs concernés à travers le Mist Request Broker.	M0004	

TAB. 5 – *Éléments de la couche Mist et leurs rôle*

sont apparues pour faire parvenir des trains au sein des zones urbaine contenant une population massive. Ceci afin de lutter contre les insuffisances dans le domaine du transport et pour que les individus puissent en bénéficier. Cependant, cela a provoqué de nombreux accidents dû à plusieurs causes souvent des inattentions de la part des conducteurs ou des piétons. A titre d'exemple, la ville de Constantine en Algérie a connu plusieurs accidents, avec des degrés de gravité distincts parmi ces accidents nous citons le déraillement du tram qui s'est encastré dans le mur de la prison du Koudiat le 16 août 2016 vers 6h05 suite a une erreur humaine. Selon les investigations ils ont constaté que le conducteur du tram a subi une crise d'hypoglycémie Boukhalfa (2016). À l'échelle mondiale, et précisément en Angleterre, le 9 novembre 2016, le tram de Londres a déraillé à Sandilands à cause d'une vitesse excessive dans un virage, 7 personnes sont décédés et 62 autre sont blessées British Transport (2016). Aussi, le 25 octobre 2018 le tram de Sheffield est entré en collision avec un camion, plusieurs personnes ont été transportées à l'hôpital, mais il n'y a pas eu de décès Taylor (2018).

Ainsi, nous concluons que la plupart des accidents sont causés par le duo "utilisateur/système", ce qui signifie que tous les événements provoquant des accidents se produisent avec l'interaction du facteur humain et du système. Il est donc essentiel de mettre en place un plan d'actions prenant en compte le facteur humain dans ces systèmes automatisés afin d'atteindre un niveau de sécurité élevé Kahlouche et Chaib (2017). En effet, des grands défis doivent être

Élément	Rôle	Code	Représentation Graphique
"Edge Gateway Router"	Permet de relier et connecter le deuxième niveau ("Mist Layer") avec le troisième niveau ("Edge Layer") pour la transmission et l'échange des données.	E0001	
"Micro Data Center"	Permet d'analyser et traiter les données provenant du deuxième niveau ("Mist Layer") pour prendre des décisions. En outre il permet le stockage et la mise en cache des données et les sécuriser.	E0002	
"Application Server"	Permet d'assurer le bon fonctionnement du troisième niveau ("Edge Layer") grâce aux applications Edge déployées. De plus, il assure une meilleure communication entre les différents nœuds Edge.	E0003	
"Cloud Gateway Server"	Permet de relier et connecter le troisième niveau ("Edge Layer") avec le quatrième niveau ("Cloud Layer"), de préparer et sécuriser les données avant les transmettre à ce dernier.	E0004	

TAB. 6 – Éléments de la couche Edge et leurs rôle

relevés par les industriels dans le monde du transport afin d'assurer la sécurité des piétons, des passagers et des véhicules durant l'itinéraire de ces trams.

5.2 Système de Contrôle des Risques

Le système de contrôle des risques ("Risk Control System") dans le transport est un système fondamental de la mobilité intelligente, conçu pour assurer la sécurité des individus dans les villes adoptant le tram comme moyen de transport. Cela, en garantissant une bonne gestion routières au sein des villes ainsi que l'augmentation des capacités technologiques des trams afin de maîtriser le contexte et éviter l'erreur humaine.

La particularité de ce travail réside dans la combinaison de ce système et le Edge Computing, ce qui engendre un nouveau système plus efficace et performant en terme de temps de latence et en bande passante. Nous proposons ainsi le système Edge pour le contrôle des risques ("Edge Risk Control System", ERCS). Ce type de système permet aux trams d'éviter plusieurs problèmes tels que le déraillement et l'écrasement des individus. En effet, la figure 8 représente l'ensemble des liens de communication entre les différents centres de traitement routiers pour échanger des informations et des données concernant l'état de l'environnement. Par conséquent, nous identifions trois éléments nécessaires qui sont : (1) les centres de traitement locaux situés au niveau du tram et des passages à niveau qui contrôlent les dispositifs aux

Architecture de Référence pour le Edge Computing

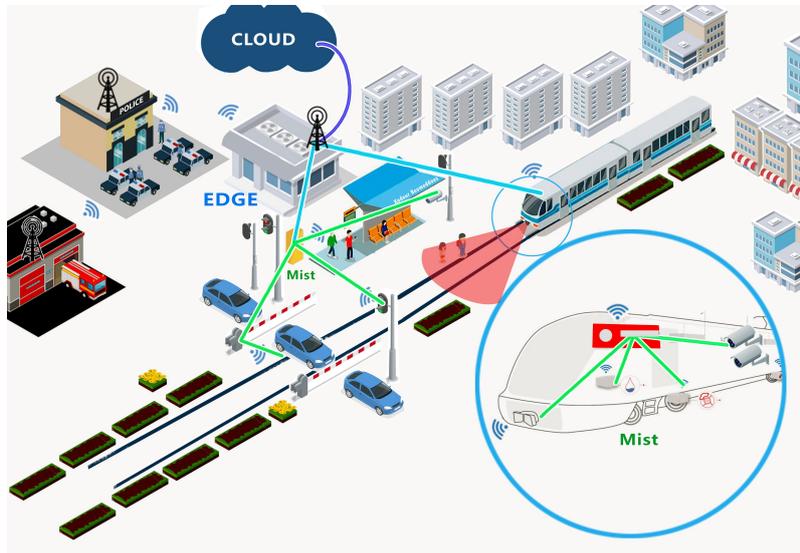


FIG. 8 – *Système Edge pour le contrôle des risques*

niveau local ; (2) le centre de traitement régional qui relie les différents centres de traitement locaux et traite leurs données en temps réel ; et enfin (3) un fournisseur Cloud. Dans ce système, nous soulignons que l'intervention de l'être humain est absente afin d'éviter les erreurs provoquées par ce dernier.

Durant son itinéraire, le tram est toujours exposé à de différents problèmes dus à des raisons techniques, des obstacles ou des événements inattendus. Ces derniers nécessitent une réaction en temps réel exigeant un temps de latence réduit. Ce qui met en évidence la proposition du jumelage entre le Edge Computing et les systèmes de contrôle des risques ("ERCs"). Pour mieux comprendre le fonctionnement du système proposé "Edge Risk Control System", nous décrivons les trois scénarios que le tram peut rencontrer :

1. Des obstacles inattendus nécessitant une réaction immédiate ce qui l'oblige de faire un traitement au niveau du centre de traitement local.
2. Des obstacles dans les passages à niveau tels qu'une voiture tombée en panne ou bien un embouteillage. Ce qui nécessite dans ce cas un avertissement préalable du tram pour qu'il soit prudent, aussi les feux de circulations doivent être ajustés pour bien gérer cette situation. Cela grâce au centre de traitement régional qui traite les données provenant du centre de traitement local du passages à niveau et envoi les résultats vers le centre du traitement local du tram.
3. Enfin, nous notons que les données de haute qualité générées par les caméras de surveillance à partir du passage à niveau seront stockées dans le Cloud due à sa capacité significative du stockage.

5.3 Instanciation de l'Architecture EECA

La figure 9 illustre l'architecture du système Edge pour le contrôle des risques ("Edge Risk Control System Architecture"). Nous identifions les quatre couches de l'architecture de référence proposée "Enhanced Edge Computing" (EECA). Elle prend en considération deux constituants : le tram ("Tram") et le passage à niveau ("Railroad Crossing"), qui sont constitués de deux couches :

- Couche "Edge Devices" représente l'ensemble des dispositifs tels que la caméra de surveillance ("Security camera"), les capteurs de distance ("Distance sensor"), la boîte noire ("Black box"), le capteur de chaleur ("AC sensor"), les freins ("Brakes"), le capteur de lumière ("Light sensor") et les feux de circulation ("traffic light").
- Couche "Mist Layer" contient l'ensemble des unités et centres de traitement locaux.

De plus, la couche "Edge Layer" représente le centre de traitement régional qui permet d'assurer le traitement des données et la coordination entre le tram et le passage à niveau. Enfin, la couche "Cloud Layer" joue le rôle d'un serveur de stockage de données. L'instanciation de l'architecture "Enhanced Edge Computing" (EECA) proposée donne une vue plus compréhensible des liens au Edge Computing, elle prend en compte les quatre couches et les interactions entre leurs différents éléments.

6 Conclusion

Le travail réalisé dans le cadre de ce papier, s'articule autour de la modélisation d'un nouveau paradigme informatique nommé : Edge Computing, qui consiste à apporter les capacités de traitement et de stockage aux extrémités logiques du réseau. Pour cela, nous avons :

1. établi une étude approfondie de l'état de l'art et les travaux existants autour de la modélisation des architectures Edge Computing.
2. proposé une architecture de référence pour le Edge Computing ("Enhanced Edge Computing") en quatre couches ("Dispositif Edge", "Mist", "Edge" et "Cloud").
3. validé notre approche à travers une étude de cas concrète en relation avec les systèmes de contrôle des risques dans le transport "Edge Risk Control System".

Les travaux réalisés dans le cadre de ce papier ouvrent plusieurs perspectives scientifiques. En particulier, nous projetons au développement d'un outil dédié à la modélisation des éléments architecturaux de l'architecture de référence "EECA" proposée.

Références

- Boukhalfa, I. (2016). Constantine : le déraillement du tramway a fait deux blessés. <http://www.liberte-algerie.com/actualite/constantine-le-deraillement-du-tramway-a-fait-deux-blesses-253164>. Accessed : 2019-04-21.
- Bradicich, T. (2018). The intelligent edge : What it is, what it's not, and why it's useful. <https://www.hpe.com/us/en/insights/articles/>. Accessed : 2019-03-12.

Architecture de Référence pour le Edge Computing

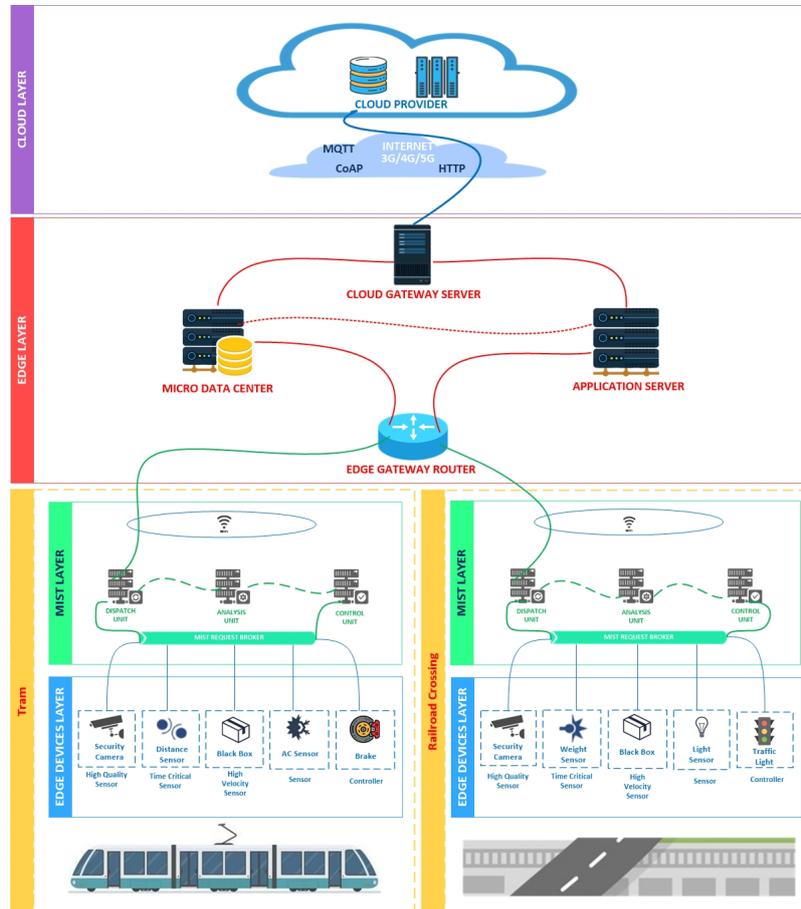


FIG. 9 – Architecture du "Edge Risk Control System"

British Transport (2016). Updated statement on tram derailment - croydon. http://media.btp.police.uk/r/13403/updated_statement_on_tram_derailment_-_croydon. Accessed : 2019-04-21.

Buyya, R., C. S. Yeo, S. Venugopal, J. Broberg, et I. Brandic (2009). Cloud computing and emerging it platforms : Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation computer systems* 25(6), 599–616.

Cao, J., Q. Zhang, et W. Shi (2018). *Edge Computing : A Primer*. Springer.

CB Insights (2019). The edge computing ecosystem : From sensors to the centralized cloud. <https://www.cbinsights.com/research/edge-computing-ecosystem/>. Accessed : 2019-03-11.

- Chang, C., S. N. Srirama, et R. Buyya (2019). Internet of things (iot) and new computing paradigms. *Fog and Edge Computing : Principles and Paradigms*, 1–23.
- De Win, B., F. Piessens, W. Joosen, et T. Verhanneman (2002). On the importance of the separation-of-concerns principle in secure software engineering. In *Workshop on the Application of Engineering Principles to System Security Design*, pp. 1–10.
- Foster, I., Y. Zhao, I. Raicu, et S. Lu (2008). Cloud computing and grid computing 360-degree compared. In *2008 Grid Computing Environments Workshop*, pp. 1–10. Ieee.
- GE Digital, Driving Digital Transformation. What is edge computing? <https://www.ge.com/digital/blog/what-edge-computing>. Accessed : 2019-03-11.
- Intel SAP (2018). Iot joint reference architecture from intel and sap.
- Jararweh, Y., F. Ababneh, A. Khreishah, F. Dosari, et al. (2014). Scalable cloudlet-based mobile computing model. *Procedia Computer Science* 34, 434–441.
- Kahlouche, A. et R. Chaib (2017). An overview of constantine s tram safety.
- Kurbel, K. E. (2008). *The Making of Information Systems : Software Engineering and Management in a Globalized World*. Springer Science & Business Media.
- Li, C., Y. Xue, J. Wang, W. Zhang, et T. Li (2018). Edge-oriented computing paradigms : A survey on architecture design and system management. *ACM Computing Surveys (CSUR)* 51(2), 39.
- Liyanage, M., C. Chang, et S. N. Srirama (2016). mepaas : mobile-embedded platform as a service for distributing fog computing to edge nodes. In *2016 17th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT)*, pp. 73–80. IEEE.
- Michael, A., F. Armando, G. Rean, D. J. Anthony, K. Randy, K. Andy, L. Gunho, P. David, R. Ariel, S. Ion, et al. (2010). A view of cloud computing. *Communications of the ACM* 53(4), 50–58.
- Preden, J. S., K. Tammemäe, A. Jantsch, M. Leier, A. Riid, et E. Calis (2015). The benefits of self-awareness and attention in fog and mist computing. *Computer* 48(7), 37–45.
- Sehout, O. N., M. Ghiat, Z. Benzadri, et F. Belala (2018). A component-based modeling of edge systems computing. In *Proceedings of the 3rd International Conference on Advanced Aspects of Software Engineering, ICAASE 2018, Constantine, Algeria, December 1-2, 2018.*, pp. 99–106.
- Sugathadasa, K. M. (2017). Distributed system architectures and architectural styles. <https://keetmalin.wixsite.com/keetmalin/single-post/2017/09/27/>. Accessed : 2019-04-15.
- Taylor, B. (2018). Sheffield-rotherham tram-train crashes after being hit by lorry. <https://www.bbc.com/news/uk-england-south-yorkshire-45976460>. Accessed : 2019-04-21.
- Trifiro, M. et J. Smith (2018a). Open glossy of edge computing : An official project of the linux foundation.
- Trifiro, M. et J. Smith (2018b). *State of the Edge : A Market and Ecosystem Report for Edge Computing*.

- Varghese, B., N. Wang, S. Barbhuiya, P. Kilpatrick, et D. S. Nikolopoulos (2016). Challenges and opportunities in edge computing. In *2016 IEEE International Conference on Smart Cloud (SmartCloud)*, pp. 20–26. IEEE.
- Villari, M., M. Fazio, S. Dustdar, O. Rana, et R. Ranjan (2016). Osmotic computing : A new paradigm for edge/cloud integration. *IEEE Cloud Computing* 3(6), 76–83.
- Yu, W., F. Liang, X. He, W. G. Hatcher, C. Lu, J. Lin, et X. Yang (2018a). A survey on the edge computing for the internet of things. *IEEE access* 6, 6900–6919.
- Yu, Z., J. Wang, Q. Qi, J. Liao, et J. Xu (2018b). *Boundless Application and Resource Based on Container Technology*, pp. 34–48.
- Zhang, B., N. Mor, J. Kolb, D. S. Chan, K. Lutz, E. Allman, J. Wawrzynek, E. Lee, et J. Kubiawicz (2015). The cloud is not enough : Saving iot from the cloud. In *7th {USENIX} Workshop on Hot Topics in Cloud Computing (HotCloud 15)*.

Summary

Edge Computing is a new paradigm that becomes the most attractive model in both industrial and academic fields. It is considered as an evolution of Cloud Computing thanks to its ability to process and store data in a decentralized way at close level of its sources and no longer on the Cloud infrastructure. Edge Computing provides a highly efficient way to improve reliability, low latency, and a good reduction in bandwidth requirements in data centers. The degree of complexity, the problem of partitioning the workload on the different layers constituting the Edge Computing architecture, and the diversity of the devices, lead to an indeterminate definition of the elements involved in Edge Computing architecture. In this paper, we implement a reference architecture dedicated to Edge Computing "EECA" (Enhanced Edge Computing) with a high level of expressiveness that supports all Edge Computing concepts. Then, we concretize the obtained results, via the implementation of a prototype regarding the use case of Edge Risk Control System (ERCS).