
Une approche basée sur les MicroÉléments pour l'Évolution des Systèmes d'Information

**Carine Souveyet¹, Massimo Villari², Luiz Angelo Steffene³,
Manuele Kirsch Pinheiro¹**

1. *Université Paris 1 Panthéon Sorbonne*
Centre de Recherche en Informatique (CRI – EA 1445)
90 rue de Tolbiac, F-75013 Paris, France
{manuele.kirsch-pinheiro, carine.souveyet}@univ-paris1.fr
2. *Università di Messina*
Future Computer Research Lab
Viale F. Stagno d'Alcontres, 31, I-98166 Messine, Italie
mvillari@unime.it
3. *Université de Reims Champagne Ardenne*
Centre de Recherche en STIC (CReSTIC – EA 3804)
Moulin de la Housse, F-51097 Reims, France
angelo.steffene@univ-reims.fr

RESUME. Des environnements intelligents tels que l'Internet des Objets (IoT), l'informatique mobile, le Cloud computing, le Fog et le Edge computing, etc.) sont de plus en plus répandus mais ne sont pas souvent considérés dans le périmètre des Systèmes d'Information (SI) traditionnels aussi bien lors de leur conception que lors de leur exécution. La mise en place de ces environnements intelligents nécessite souvent une attention sur des problématiques techniques et des configurations spécifiques, ce qui rend difficile leur intégration et leur gestion au sein de ces systèmes. Cet article propose une abstraction conceptuelle de la complexité et l'hétérogénéité technique, grâce au concept de microéléments associés à des ressources types afin d'exprimer les services qu'un environnement intelligent peut apporter au SI pour améliorer les processus métiers existants. L'approche préconisée permet d'étendre le Système d'Information traditionnel par de nouveaux processus métiers, exploitant le potentiel de ces environnements intelligents. L'abstraction conceptuelle proposée a l'avantage (1) d'exposer au niveau de l'entreprise les services fournis par l'environnement technique intelligent ; (2) d'exprimer au niveau conceptuel les configurations informatiques requises pour exécuter plus facilement les processus métier ; et (3) de relier le niveau conceptuel et le niveau technique sur les concepts de microéléments au niveau conceptuel et de microservices au niveau technique.

Mots-clés : Microélément, microservices, système d'information.

1. Introduction

Les derniers développements informatiques tels que l'Internet des Objets (Gubbi et al., 2013), le *Cloud computing* (Mell & Grance, 2011), ou le *Fog/Edge computing* (Chen et al., 2017) ont permis l'apparition d'environnements informatiques intelligents qui sont complexes, hétérogènes et souvent mobiles. Ces environnements techniques sont difficiles à gérer mais, en même temps, ils peuvent être utilisés afin d'étendre les capacités des applications métiers de manière à ce qu'elles soient plus répandues, ubiquitaires et flexibles.

Malheureusement, les solutions conceptuelles utilisées dans la communauté SI, souvent orientées services et/ou processus métier, ne tiennent pas réellement compte des couches techniques (Kaabi & Souveyet, 2007 ; Hinkelmann et al., 2016 ; Sanchez-Puchol et al., 2017 ; Neumann et al., 2014). Ce fossé entre la vision conceptuelle d'un SI et la vision technique de celui-ci est déjà reconnu par la littérature (e.g. Kaabi & Souveyet, 2007 ou Trabelsi, 2014). L'évolution des ressources grâce à l'intégration d'environnements pervasifs n'a fait qu'aggraver cette situation. Or, des nouvelles opportunités pour les entreprises et leur processus métiers sont aujourd'hui laissées de côté car ces nouvelles ressources (pourtant utilisées pour le déploiement des services et applications) demeurent non intégrées aux SI ou, en tout cas, à son management. La prise en compte de l'environnement technique ouvre en effet des perspectives pour les processus métiers qui peuvent exploiter désormais les nouvelles données et modes d'interaction proposés par ces environnements intelligents.

L'objectif de cet article est ainsi d'entamer une réflexion sur les stratégies permettant d'assimiler au niveau conceptuel une vision de l'environnement technique d'un Système d'Information. Nous proposons la conceptualisation de l'environnement technique au niveau métier à travers une abstraction de *business container*, laquelle permet de mieux appréhender l'environnement technique et ses ressources au niveau conceptuel et ainsi pouvoir étendre le processus métier traditionnel en fonction des éléments et des informations disponibles sur cet environnement.

Cet article est organisé comme suit : la Section 2 résume le traitement accordé aux environnements techniques dans la communauté SI au vu des nouvelles opportunités offertes par les nouvelles technologies. La Section 3 présente notre proposition, qui est illustrée par un scénario à la Section 4, avant de conclure à la Section 5.

2. État de l'Art

Pendant des années, la notion de ressources n'était pas considérée comme stratégique lors de la modélisation et de la conceptualisation d'un SI. Cela s'explique en grande partie par l'homogénéité des ressources dans les Systèmes d'Information traditionnels (Kirsch Pinheiro & Souveyet, 2019). Ceci a changé depuis l'introduction du *Cloud computing* et plus récemment de l'IoT et du *Fog/Edge computing*, lesquels offrent des nouvelles opportunités de travail, autant au niveau technique, qu'au niveau métier. Derrière ces nouvelles technologies et ressources viennent des nouveaux services et des données qui peuvent désormais alimenter le SI.

Cependant, l'étude des SI et de leur gestion fait souvent abstraction de l'environnement technique sur lequel repose l'exécution de ces systèmes. Même si la littérature mentionne l'existence d'un fossé entre le niveau conceptuel et le niveau technique sur différents aspects de ces systèmes (Kaabi & Souveyet, 2007 ; Perini et al., 2005 ; Mirbel & Crescenzo, 2010 ; Trabelsi, 2014), la prise en compte de ce dernier sur le premier s'avère particulièrement limitée. Bien souvent, il n'y a pas de lien direct entre les deux niveaux, ou il n'y a pas de *feedback* permettant d'alimenter le niveau conceptuel avec des informations issues de l'environnement d'exécution. Pourtant, des nouvelles approches ont été proposées afin de tenir compte des nouvelles évolutions technologiques (e.g., Neumann et al., 2014 ; Hinkelmann et al., 2016 ; Sanchez-Puchol et al., 2017 ; Rao et Angelov, 2009). La prise en compte de l'environnement technique au niveau conceptuel demeure limitée et bien souvent dans une démarche presque exclusivement descendante : la vision business peut être projetée sur le niveau technique, mais celui-ci n'a pas de vraie existence conceptuelle. Les informations concernant l'état de cet environnement en cours d'exécution ne sont que rarement remontées au niveau métier.

Ce manque de vision, au niveau conceptuel, de la réalité au niveau technique a des conséquences pour les SI, puisque ceux-ci se trouvent privés de nombreuses opportunités d'évolution.

3. Proposition

L'objectif de cet article est de proposer une manière innovante et flexible d'étendre la vision traditionnelle du Système d'Information (SI) afin d'intégrer l'utilisation et la configuration des environnements techniques intelligents. Nous pensons qu'une meilleure intégration des environnements techniques intelligents (IoT, informatique mobile, *Cloud computing*, *Fog/Edge computing*, etc.) aux SI est essentielle pour assurer l'évolution de ces systèmes. Cette intégration se fait à partir du point de vue métier et intervient aussi bien durant la conception que l'exécution du SI.

Différents challenges s'opposent à cette intégration. Tout d'abord, nous avons des SI qu'on peut appeler "traditionnels", qui sont solidement ancrés dans les organisations et qui proposent déjà de nombreux processus métier. Toute évolution de ces systèmes doit se faire en respectant cet héritage, qui doit être enrichi sans être remis en question.

Or, les environnements techniques, devenus désormais intelligents, offrent des nouvelles possibilités à ces processus existants et de manière générale, de nouveaux moyens de satisfaire des objectifs métier. Pour cela, il faut que cet environnement technique, avec les services et données qu'il offre, soit visible au niveau métier. Il faut donc assurer à cet environnement réel une existence conceptuelle permettant au métier de percevoir les opportunités offertes. Inversement, les nouveaux processus qui peuvent découler de cette prise en compte de l'environnement doivent être projetés à partir du niveau métier (conceptuel) vers le niveau technique pour leur exécution.

L'interaction entre le niveau conceptuel et l'environnement technique doit ainsi se faire dans les deux sens : en remontant des informations et services présents dans

4 Atelier Évolution des SI : vers des SI Pervasifs ?

l'environnement au niveau métier, et en projetant les éléments business de leur niveau conceptuel jusqu'à leur exécution sur l'environnement technique réel. Nous abordons cela à l'aide d'un modèle d'abstraction simplifié en deux niveaux qui représente le niveau technique et le niveau conceptuel des SI. En passant en revue leurs interconnexions, nous cherchons à proposer une instance d'exécution capable de vérifier l'exactitude de ce qui a été envisagé au niveau de la conception.

Cette approche, basée sur une approche par conteneurs, propose de décrire conceptuellement l'environnement technique et sa perspective business en deux types de conteneurs inter-reliés. D'un côté, nous proposons un modèle de conteneurs business, appelés *micro-business containers* (μ BC), lesquels agissent comme une abstraction au niveau conceptuel d'un élément business réalisable par le SI à son niveau technique. Ce conteneur business représente ainsi une *unité métier*, avec ses règles, ses services et ses informations. De l'autre côté, des conteneurs techniques effectuent la mise en œuvre des conteneurs business, notamment sous la forme de microservices, lesquels peuvent être déployés sur différents types de ressources. La connexion entre ces deux niveaux est assurée grâce à la notion de microélément, une abstraction originellement proposée par Carnevale et al. (2018). Les microéléments sont intégrés au conteneur business, et à leur tour sont associés à des microservices du niveau technique. Ainsi, ces unités business peuvent être aisément projetées sur l'environnement technique, et inversement, les microservices et les ressources présents dans cet environnement peuvent être progressivement remontés et intégrés au niveau business. L'usage des *micro-business containers* et des microéléments offre ainsi une perspective business à l'environnement technique lequel se trouve ainsi visible au niveau stratégique de gestion d'un SI.

Les *micro-business containers* (μ BC) offrent une vision sur les capacités de l'environnement technique, grâce notamment aux microéléments. Ces conteneurs business peuvent ainsi s'intégrer au SI existant (voir Figure 1(a)), permettant d'enrichir celui-ci par l'apparition de nouveaux processus métier ou en ajoutant de la variabilité dans les processus existants. Nous détaillons chacun de ces éléments (*micro-business containers* et microéléments) dans la section suivante.

3.1 Les Micro-Business Containers (μ BC)

Comme nous l'avons expliqué précédemment, l'objectif des conteneurs business est de proposer une abstraction au niveau métier des capacités offertes par l'environnement d'exécution et, en même temps, de permettre la projection sur ce dernier d'éléments métiers explorant ces capacités afin de satisfaire les besoins propres au Système d'Information. Pour cela, les conteneurs business doivent abstraire les technologies disponibles dans l'environnement technique à travers les services et les données offertes ou accessibles à travers ces technologies. Ils doivent également offrir une abstraction des ressources intégrées à cet environnement, puisque comme il est souligné par Kirsch-Pinheiro & Souveyet (2019), ces ressources relèvent aujourd'hui d'un choix stratégique pour les SI.

À partir de ces constats, nous proposons la notion de *micro-business containers* (μ BC), illustrés par la Figure 1(b). Celle-ci est composée de 5 éléments majeurs permettant la description de différents aspects nécessaires à cette notion.

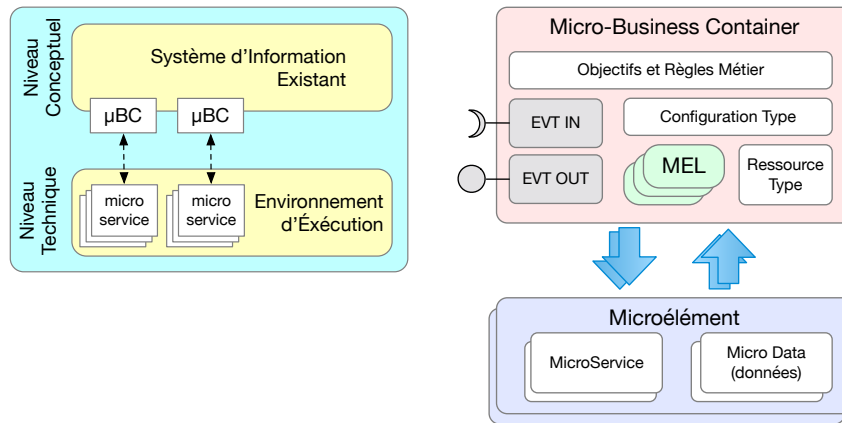


Figure 1 – (a) Le rôle de liaison des micro-business containers, et (b) Composition d'un micro-business container et sa relation avec les microéléments

Tout d'abord, il nous paraît essentiel de pouvoir remonter à un niveau conceptuel l'ensemble des services et données présents au niveau technique. Ces deux éléments sont encapsulés au sein des *microéléments* (cf. Section 3.2), permettant ainsi de représenter des services mis en œuvre par des microservices au niveau technique, et les données manipulées ou accessibles par ces services.

Ces services doivent s'exécuter sur des ressources disponibles dans l'environnement. Ces ressources étant de plus en plus hétérogènes et mobiles, la description au niveau conceptuel de chaque ressource réellement disponible devient impraticable. Nous proposons alors que les ressources soient décrites par un *profil type*, où les différents types de ressources seront identifiées en fonction des capacités attendues, sans que leurs instances soient clairement fixées.

L'instanciation de ces ressources type vers des ressources réelles doit se passer en temps d'exécution suivant la disponibilité de ces ressources et les consignes indiquées par le conteneur business dans sa *configuration type*. Celle-ci indique les contraintes nécessaires pour le placement de ces services sur les ressources, comme par exemple le nombre d'instances souhaitées.

Par ailleurs, ces services répondent à des besoins métiers, des *objectifs* qui doivent être atteints par le SI et ses acteurs. Ces besoins peuvent être représentés à l'aide de buts, à l'instar de Najar et al. (2011) ou Mirbel & Crescenzo (2009), qui associent à un service un ensemble de buts métiers dans une approche intentionnelle.

Enfin, les services et les données encapsulés par les microéléments doivent s'intégrer au SI déjà en place, pouvant ainsi participer à différents processus métiers.

L'interaction entre les conteneurs business et les processus définis au sein de ce système se passe ainsi à travers un ensemble d'évènements produits et/ou attendus par le conteneur. En effet, un conteneur business peut produire des évènements attendus et observables par d'autres composants du Système d'Information dans le cadre d'un processus métier. Inversement, pour participer pleinement à un processus métier, le conteneur business peut observer des évènements issus d'autres composants du système. Ces évènements peuvent également guider l'orchestration de plusieurs conteneurs business au sein des nouveaux processus intégrés au système.

Le *micro-business container* se présente ainsi comme une abstraction permettant, d'une part, de visualiser les éléments présents dans l'environnement technique au niveau métier. Il s'agit, dans ce sens, d'une projection business de l'environnement d'exécution. D'autre part, ils permettent la représentation des services attendus par la couche métier sur la couche technique.

3.2 La notion de Microélément

Proposé par Carnevale et al. (2018), un microélément (aussi appelé MEL pour MicroElement) est composé par deux éléments abstraits, cf. la Figure 2 : les MicroServices (MS) et les MicroData (MD). À leur tour, ces deux composants abstraits donnent origine à des instances Utilisateur (MUD pour MicroUserData et MUS pour MicroUserService) ou Système (MOS pour MicroOperationalService et MOD pour MicroOperationalData). Les flèches fines de la Figure 2 montrent cette composition hiérarchique, alors que la grande flèche indique une dépendance entre les MicroServices (MS) et les MicroData (MD).

Le volet *Operational* représente un aspect intégrant l'environnement (comme par exemple un *daemon* du système d'exploitation et sa configuration) alors que le volet *User* indique des éléments situés dans l'espace utilisateur du système d'exploitation (e.g. une application et les données utilisateur).

L'intérêt de cette approche est qu'elle permet de détacher certains éléments, vu que par définition le MD et le MS sont mobiles, portables et multiplateformes. Ainsi, un exemple de MS serait un conteneur Docker, attaché à un MD contenant des métadonnées en format JSON utilisées pour le provisionnement du conteneur, ou des données stockées dans un volume Docker accessible aux conteneurs.

De plus, les MD peuvent être autant passives (des données qui peuvent être lues ou mise à jour directement à partir de dispositifs de stockage grâce à des appels système) ou actives, comme par exemple les données issues d'une application de bases de données. En effet, dans une base de données, le contenu et le format final des données sont définis en temps d'exécution par les requêtes, contrairement à des données statiques lues à partir d'un fichier.

Plus exactement, un microélément et ses composants permettent une transition graduelle entre la représentation de l'information au niveau conceptuel et les éléments techniques qui effectuent la capture de l'information ou la mise en place des fonctionnalités requises. Cette structuration permet entre autres un couplage faible

entre le *micro-business container* (μ BC) et instances de microservices utilisées, donnant plus de flexibilité à l'ensemble. Non seulement la composition de microservices dans le but de respecter le profil-type des μ BC peut se faire dynamiquement (par exemple, lors de la sélection d'une caméra destinée à la vidéosurveillance d'un bâtiment), comme l'information remontée par les microservices peut être adaptée avant son intégration au SI : agrégation de différentes sources de données IoT, identification d'événements, génération d'alertes ou de rapports, etc.

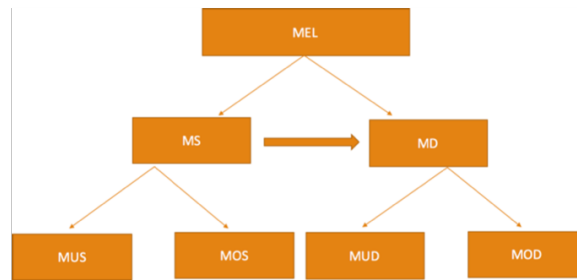


Figure 2 – Architecture des MicroELeMents (Carnevale et al., 2018)

4. Scénario illustrant la solution proposée

Le scénario proposé est celui de *l'accueil de Monsieur Rossi professeur invité dans une université parisienne en provenance d'Italie*. Le système d'information de cette université (IS@Paris) a déjà enregistré les données relatives aux processus administratifs autour de la gestion des professeurs invités comme la validation de la demande d'invitation du prof. Rossi effectuée par un enseignant local et la préparation du voyage et du séjour du prof. Rossi. A la fin de cette préparation, nous connaissons la date d'arrivée du professeur invité et le bureau qui lui a été affecté pendant toute la durée du séjour par l'invitant et l'administration. Néanmoins à son arrivée, il faut qu'il se déplace au service de gestion des invités pour attester de sa présence au sein de l'université. De plus, il doit se déplacer au service technique pour que son ordinateur portable soit configuré afin d'accéder aux ressources (imprimantes, etc.) du bureau utilisé durant son séjour.

Le système envisagé, illustré en Figure 3, consiste à étendre IS@Paris en ajoutant une nouvelle procédure d'accueil où la présence du prof. Rossi est détectée de manière transparente dès lors son authentification au réseau *Eduroam* au sein de l'université. Cela lui évite d'avoir à se déplacer au service administratif. Le confort du prof. Rossi est aussi amélioré par le fait que la configuration de son ordinateur portable lui sera proposée automatiquement, permettant d'accéder au service d'impression et de numérisation disponible dans le bureau qui lui a été attribué. Cette extension sera intégrée au processus initial comme une alternative possible aux tâches « accueil service gestion » et « configuration salle ».

Pour la réalisation de cette nouvelle procédure d'accueil, il faut un microélément d'authentification spécifique à *Eduroam* détectant la première connexion d'un professeur invité et un second microélément pour la configuration de l'ordinateur portable de l'invité en fonction des périphériques existant dans le bureau qui lui est affecté. Ces deux microéléments sont ainsi encapsulés au sein d'un même conteneur business destiné à l'accueil d'un invité. Celui-ci est intégré à la gestion des invités à côté de la procédure d'accueil existante, proposant ainsi une nouvelle procédure qui s'ajoute à la précédente. Il est à noter que le conteneur métier permet au microélément d'accéder aux données du SI pour savoir, par exemple, si l'utilisateur qui se connecte est un invité attendu qui se connecte pour la première fois mais aussi pour déterminer la configuration en impression et numérisation du bureau affecté à cet invité.

Cet exemple montre les deux sens possibles d'usage des microéléments ; une action est détectée au niveau de l'environnement technique et remontée au niveau du conteneur business pour produire un événement métier d'arrivée d'un professeur invité dans les locaux de l'université et l'autre est l'activation du microélément au niveau métier qui va agir en invoquant le microservice technique permettant la configuration de l'ordinateur portable de l'invité. Le processus d'accueil modifié utilise le conteneur business car il consomme l'événement d'arrivée d'un professeur invité et déclenche une activité automatique qui va noter dans le SI que le professeur invité est bien arrivé et déclenche la configuration de son ordinateur portable en invoquant le microélément de configuration exposé par le conteneur business.

Cet exemple basique montre l'intégration de la couche métier et de la couche technique dans les deux sens d'interaction. Néanmoins, il faut descendre au niveau de la couche technique pour savoir quel dispositif réel l'université a mis en place pour les services d'authentification et de configuration. Sans connaissance au niveau métier de ces services et de ces ressources (authentification *Eduroam*, microservice d'impression, etc.), le scénario devient difficilement envisageable, et les améliorations apportées au confort des utilisateurs (ici le prof. Rossi) peuvent ainsi être perdues.

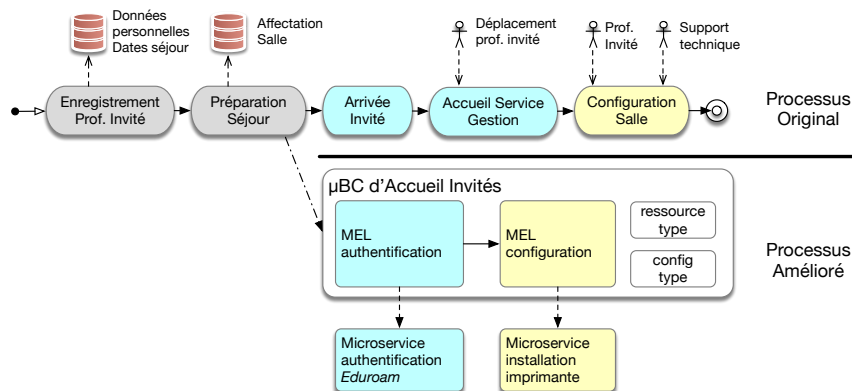


Figure 3 - Processus invité traditionnel versus processus amélioré par les μBC

Nous ne traitons pas dans ce papier comment le modèle de processus peut formellement intégrer les services du micro business container ou consommer les événements produits par celui-ci.

5. Conclusions

Dans cet article nous proposons une nouvelle vision pour la prise en compte de l'environnement technique au niveau conceptuel. Cet environnement, ayant beaucoup évolué ces dernières années, se présente de plus en plus comme un environnement intelligent offrant des nouvelles perspectives pour les processus métier. Nous soutenons ici la nécessité d'une abstraction permettant l'intégration des capacités offertes par cet environnement au sein d'un Système d'Information traditionnel.

Nous proposons pour cela la notion de *micro-business container* (μ BC), laquelle représente une abstraction de l'environnement technique permettant d'enrichir les SI traditionnels par l'intégration de nouveaux éléments aux processus déjà en place sur le système. À travers cette notion, nous cherchons à projeter l'environnement technique à un niveau conceptuel et ainsi pouvoir étendre les processus métier afin de tirer profit des services et des données disponibles sur l'environnement technique. Inversement, la présence de cette abstraction permet d'envisager la remontée de l'information en temps d'exécution. Il serait possible ainsi d'alimenter le niveau métier avec des informations issues de l'environnement technique en temps réel. Ces informations pourraient permettre une meilleure gestion des processus métiers, de leur exécution, voire de leur optimisation.

Cette proposition n'est qu'à son début et de nombreux travaux sont à prévoir. Tout d'abord, la définition d'un descripteur détaillant les μ BC et ses composants, et notamment la définition des ressources type et de leurs configurations. Les possibilités d'orchestration des μ BC à travers les événements doivent également faire l'objet de travaux futurs, dont l'intégration de ces conteneurs dans un environnement d'exécution au niveau métier. Un tel environnement est nécessaire afin d'associer la remontée des informations à partir du niveau technique, mais également afin de permettre le pilotage de ce dernier à partir du niveau métier. Ce pilotage implique aussi la gestion de la mise en correspondance entre le conteneur business, les microéléments, les microservices et les ressources présentes dans l'environnement technique.

Références

- Carnevale, L., Celesti, A., Galletta, A., Dustdar, S., Villari, M. (2018). From the Cloud to Edge and IoT: a Smart Orchestration Architecture for Enabling Osmotic Computing. *32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, Krakow, pp. 419-424.
- Chen, S., Zhang, T., Shi, W. (2017). Fog computing, *IEEE Internet Computing*, vol. 21, p 4-6.
- Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions, *Future Generation Computer Systems*, vol. 29, n° 7, p. 1645-1660.

- Hinkelmann, K., Gerber, A., Karagiannis, D., Thoenssen, B., van der Merwe, A., Woitsch, R. (2016). A new paradigm for the continuous alignment of business and IT: Combining enterprise architecture modelling and enterprise ontology, *Computers in Industry*, vol. 79, June 2016, p. 77 – 86.
- Kaabi, R. S., Souveyet, C. (2007). Capturing intentional services with business process maps. In: C. Rolland, C.; Pastor, O. & Cavarero, J. (Eds.), *1st IEEE International Conference on Research Challenges in Information Science (RCIS)*, IEEE, p. 309-318.
- Kirsch Pinheiro, M., Souveyet, C. (2019). Le Rôle des Ressources dans l'Évolution des Systèmes d'Information. *37^{ème} Congrès INFORSID (INFormatique des ORganisations et Systèmes d'Information et de Décision)*, Juin 2019, Paris. A paraître.
- Mell, P., Grance, T. (2011). The NIST definition of Cloud computing, *National Institute of Standards and Technology (NIST)*, Tech. Rep. 800-145.
- Mirbel, I., Crescenzo, P. (2010). From end-user's requirements to Web services retrieval: a semantic and intention-driven approach. In: J.-H. Morin, J. Ralyte, M. Snene (Eds.), *Exploring service science, First International Conference (IESS 2010)*, LNBIP, vol. 53, Springer, p. 30-44.
- Najar, S., Kirsch-Pinheiro, M., Souveyet, C. (2011). The Influence of Context on Intentional Service. *IEEE 35th Annual Computer Software and Applications Conference Workshops (COMPSACW)*, p. 470-475.
- Neumann, G., Sobernig, S., Aram, M. (2014). Evolutionary Business Information Systems. *Business & Information Systems Engineering*, vol. 06, p. 33-38.
- Perini, A. Susi, A., Mylopoulos, J. (2005). Tropos Design Process for Web Services, *Proc. 1st Int. Workshop Service-Oriented Computing: Consequences for Eng. Requirements*, Sept. 2005.
- Rao B., Angelov B. (2009). Pervasive Information Systems Value Chain: A Services Perspective. *International Journal of Innovation and Technology Management*, vol. 06, n° 01, p. 17-40.
- Sanchez-Puchol, F., Pastor-Collado, J. A., Borrell, B. (2017). Towards an unified Information Systems reference model for higher education institutions. *Procedia Computer Science*, vol. 121, p. 542 – 553
- Trabelsi, L. (2014). Les Systèmes d'Information Urbanisés : étude et analyse de leur performance. Thèse de doctorat en Sciences de Gestion. Université Nice Sophia Antipolis, 2014.