A Peer-to-Peer Model for Virtualization and Knowledge Sharing in Smart Spaces

Dmitry G. Korzun
Department of Computer Science
Petrozavodsk State University (PetrSU)
Petrozavodsk, Russia
e-mail: dkorzun@cs.karelia.ru
Helsinki Institute for Information Technology (HIIT)
and Department of Computer Science and Engineering (CSE)
Aalto University
Helsinki, Finland

Sergey I. Balandin
ITMO National Research University
St. Petersburg, Russia
FRUCT Oy
Helsinki, Finland
e-mail: sergey.balandin@fruct.org

***Abstract*—This paper is targeted to initiate discussion on how to describe and make formal definition of smart spaces in Internet of Things (IoT) environment by utilizing well-known models for Peer-to-Peer (P2P) networks. Indeed, when one starts studying smart spaces solutions and applying them in IoT environments then traditional models of P2P interaction come in mind as the first association. Every device, sensor, or network process could be represented by the corresponding agent (or peer). Services are emerged as a result of cooperative work of multiple agents in the smart space. Each agent contributes to the service by sharing its portion of knowledge in the smart space. We propose initial ideas on how a P2P model can virtualize physical objects and service construction processes by representing them as a network of interacting information objects in the smart space. Although interaction between objects is not physically direct communication, the model logically organizes direct interaction of objects as peers. This approach aims at higher interoperability in knowledge sharing and at an effective abstract level for service design.**

***Keywords*–Smart spaces; Peer-to-Peer; P2P; Multi-agent; Internet of Things; IoT; Services.**

I. INTRODUCTION

We are about to see how hundreds of billions of interconnected devices envisioned by the Internet of Things (IoT) will finally become part of our daily life. Service networks build on IoT technology is become a reality of today and a strong call for making proper models and analysis of such networks. It is important to keep in mind that in the new generation of service networks most of communications will be handled between machines without direct contribution to some particular user service. Already, we see that Machine-to-Machine (M2M) communications are gaining momentum and soon will be mature enough to take significant share of real-world applications in various fields of our life.

Generally, IoT environments are becoming large, highly dynamic, hyperconnected, and functionally distributed, e.g., see [1]–[5] and references therein. Typically, an IoT environment consists of multiple heterogeneous networks with a large number of networked elements and users' devices. Further evolution of the IoT concept envisions increasing of the number of connections by yet another order of magnitude from currently connected approximately 10 billion "things". This will result in unprecedented challenges in network scalability, resource efficiency, privacy considerations, and overall management of this multitude of "things". The traditional models of networks organization would have serious problems to deal with it, so more and more often some alternative ways to network virtualization are considered [5].

Another key trend that we witness nowadays is a demand for making services be proactive and smarter to increase efficiency of IoT environment use and free more time for the user. Along this trend, over the past few years, we have seen many predictions and comments on importance and future perspectives of the smart spaces paradigm [4][6]–[9]. Despite of its elegance and clear advantages, we must admit that the paradigm still has very limited practical use. One of the problems is that its model of virtualization and knowledge sharing is still not so clear for service developers. On the other hand, we can see that these models are very close and even similar to what has been applied for many years in the Peer-to-Peer (P2P) systems area.

This study elaborates on how to apply models for virtualization and knowledge sharing in smart spaces deployed in IoT environment. We focused on the traditional approach to modeling P2P networks [10]. Our intention is to see how such models can be adopted for the problems of knowledge virtualization and sharing. As we know it is not a trivial task to make a useful model for the considered problem. In this paper, we are not constructing a finalized ready model that answers most of the questions, rather we are sharing results of study and analysis on how to adopt well-known P2P models for the emerging application area of smart spaces.

The rest of the paper is organized as follows. Section II states the problem of knowledge representation for smart spaces. Section III presents related work and enabler approaches to modeling for knowledge representation in the smart spaces area. Section IV describes our initial P2P model for knowledge virtualization and sharing. Section V discusses the use of P2P model for service construction and delivery. Section VI summarizes the paper.

II. SMART SPACES

Let us study specific features of smart spaces deployed in localized IoT environment. Such an environment consists of surrounding IoT devices (embedded in the physical environment or appeared as mobile entities), communication network that connecting them, plus it has access to the global Internet

 ISBN: 978-1-61208-353-7

with its diversity of services. Later we focus only on smart spaces that belong to this category.

In general, the smart spaces paradigm aims at development of ubiquitous computing environment, where participating entities acquire and apply knowledge to adapt services to the inhabitants in order to enhance user experience, quality and reliability of the provided information [6]–[8]. A primary operational element is a smart object—an autonomous information processing unit.

The term "smart" means [11] that the object is (i) active, (ii) digital, and (iii) networked. Any smart object (iv) operates to some extent autonomously, (v) is re-configurable, and (vi) has local control of resources it needs to utilize (e.g., energy and data storage). The IoT concept supports this vision on smart objects [2]. The most common view of IoT refers to the connection of physical objects, while the core of technology is in information interconnection and convergence. Operation of IoT solutions is based on continued processing of huge number of data flows, originated from various sources and consumed by multiple applications.

In contrast to this basic IoT vision, a smart object in the smart space is not necessarily attached to a fixed device, as any available device can host the object. This kind of virtualization provides a powerful abstraction for creating complex systems. For instance, the M3 concept for smart spaces employs the term "knowledge processor" (KP) to emphasize the processing responsibility of each object [4][12]. Services are constructed as interactions of smart objects in this shared space. The deployment flexibility is very high. For example, the smart space can be deployed using a cloud or on user's devices that interact with each other and use pertinent services regardless of the physical location.

The M3 concept further evolves this IoT-based fusion of physical and information worlds [12]. M3 stands for Multidevice, Multidomain, and Multivendor. An M3 smart space makes it possible to mash-up and integrate information between a wide spectrum of applications and domains spanning from embedded domains to the Web. Information from physical world (objects and devices in the physical environment) becomes easily available for participants in the shared smart space. The latter also is a hub linking the information to other services and solutions in the Internet. Therefore, smart spaces open embedded data kept in many surrounding devices to use by applications for creating local services in various physical places [8].

The multitude of participants (humans, machines, processes) obviously leads to the interoperability problem. The M3 concept provides the following conceptual solution. Ovaska *et al.* [7] defined a smart space a digital entity where the relevant real-world information (i.e., information about the physical environment, the objects therein located, and the recent situation) is stored in an interoperable, machine understandable format, kept up to date and made available to unanticipated and authorized situation dependent applications. Resource Description Framework (RDF) format from the Semantic Web provides a proper representation model to store the shared information [13][14]. SPARQL (Simple Protocol and RDF Query Language) is a query language to effectively retrieve and manipulate the information in the RDF format.

This definition supports three interoperability levels [8].

1) At the bottom, the communication level provides techniques for transmitting data between devices. It enables the device and network world to exchange bits.
2) At the middle, the service level provides technologies for devices to share services in the smart space. It enables the service world to use the services across device boundaries.
3) At the top, the information level allows the information to be understood similarly in all the smart objects. It equips the information world with the interoperability means to make the same meaning of information for different participants.

The notion of semantics is subject to various definitions, e.g., see Aiello *et al.* [15]. Since a smart space aims at encompassing (directly or indirectly) all information pieces the application system needs for service operation, we can characterize semantics as follows. Semantics is a relationship or mapping established between such information pieces. This definition also covers the case when relations are established implicitly, due to relating elements of the information structure. For instance, in ontology terms, such implicit relations appear between concepts (classes).

III. Related Work and Enabler Approaches

Let us discuss existing research on approaches to modeling for virtualization and knowledge sharing applicable in the case of smart spaces. The considered approaches will be adopted in the proposed P2P model later in Sections IV and V.

Halevy and Madhavan [16] introduced the corpus-based representation principle for large collections of knowledge fragments. Unlike a traditional knowledge base with careful ontological design, a knowledge corpus consists of independent uncoordinated contributions. This idea suits well to smart spaces where many autonomic participants share information and apply the collaboratively collected knowledge.

Bertossi and Bravo [17] considered virtual integration of many different data sources. A mediator (software system) offers a common interface to a set of autonomous, independent and possibly heterogeneous data sources. The same approach is applicable for organization of smart spaces content. The primary data are kept in their sources. The smart space acts as an informational hub to relate all the data and to provide to participants a single access point.

Patouni *et al.* [5] summarized recent virtualization trends for IoT environments development. The dynamics of such a hyperconnected and full of data telecommunications environment need moving the functionality to the network edges. For this purpose logical network services are distinguished from physical resources. Furthermore, Software Defined Networks (SDN) propose decoupling of the network control and data planes, moving the control of the network behavior to third party software. The idea is similar to solutions applied in P2P based large-scale network infrastructures [10]. Compared to our case, physical entities and resources are virtually represented in smart spaces, and the appropriate smart space supports making control decisions.

Aiello *et al.* [15] discussed the notion of emergent semantics. Local semantics from information agents are consolidated into a global, population-wide semantics. Knowledge representation structures emerge from continuous interaction of the agents. This incremental, bottom-up, semi-automatic

 ISBN: 978-1-61208-353-7

construction follows a P2P style, without relying on pre-existing, global semantic models. Emergent semantics supports virtual data integration in smart spaces: data of already existing sources may be updated, added, or deleted; new sources and services may appear and disappear dynamically.

Gorodetsky [9] studied smart space generic architecture composed of many agents interacting as peers in a P2P system. The agents are mediators for data integration in the smart space as well as they take care about construction of smart services and their delivery to users. The P2P approach is used for structuring agent interaction, i.e., establishing relations between agents for direct communication.

Pellegrino *et al.* [18] proposed a P2P-based infrastructure for distributed RDF storage and a publish/subscribe layer for storing and disseminating RDF events. The P2P approach allows constructing a large-scale distributed system for knowledge sharing based on existing Semantic Web technologies.

Matuszewski and Balandin [19] presented a P2P model and system architecture for knowledge sharing in mobile environments. Humans are treated as peers. Their collective knowledge is arranged into a distributed hierarchical structure based on user-defined relations between objects and references to the data sources of other peers.

IV. CONTENT REPRESENTATION MODEL

A characteristic property of any smart space is information sharing with knowledge self-generation from the collected content [4][6][9]. Ideally, all data a service needs should be accessed via its smart space: either the data are directly stored in the smart space or they are accessed indirectly by a kept reference. The property leads to many concurrent and low coordinated contributions, and we can consider information content of a smart space a large dynamic collection I of disparate knowledge fragments.

No careful design of a single comprehensive ontology or a database schema in advance is possible to represent finely tuned structure of the content. The corpus-based representation principle is used instead [16]. Smart space content I is structured dynamically, in ad-hoc manner. For its participants, the smart space provides search query interfaces to reason knowledge over I and its instant structure.

Based on the ontological modeling approach, we can consider I consisting of information objects and semantic relations among them [4][7][14]. Its basic structure is defined by problem domain and activity ontologies (classes, relations, restrictions), e.g., using the Web Ontology Language (OWL) from the Semantic Web. Factual objects in I are represented as instances (OWL individuals) of ontology classes and their object properties represent semantic relations between objects.

The well-known P2P approach [10] can be applied for modeling the virtualization of objects in the smart space and the derived knowledge representation. Any object $i \in I$ is treated a peer. Each i keeps some data (values of data properties) and has links to some other objects j (object properties). Therefore, a P2P network G_I is formed on top of I. Contributions from smart space participants (insert, update, delete) change the network of objects, similarly as it happens in P2P due to peers churn and neighbors selection. We shall also use the terms a node and a link when referring an element in G_I and its relation.

This P2P model extends the notion of ontology graph (interrelated classes and instances of them) to a dynamic self-organized system. The following model properties clarify this extension and show the role of enabler approaches from Section III.

Virtualization: Objects in G_I are self-contained pieces of information. It can be effectively described using OWL in terms of individuals and classes. Each object provides a digital representation of a real thing (sensor, phone, person, etc.) or of an artificial entity (event, service, process, etc.). This property suits well the IoT concept as well as its evolution to Internet of Everything [5]. Participants (agents) and information objects become equal nodes. From the point of view of applications, all essential system components become observed on “one stage” (with all semantic relations) and manipulated by changing their information representation (digital).

Hierarchy: The decomposition principle from ontological modeling allows defining semantic hierarchies of concepts, e.g., hierarchy of classes of an ontology. Objects in G_I becomes connected with hierarchical semantic links, as it happens in hierarchical P2P systems. In particular, this idea was applied by Matuszewski and Balandin [19] for P2P-like structuring personal information about a person and groups of persons.

Emergent semantics: There can be non-hierarchical semantic relations in G_I. They reflect the recent state of the dynamic system. For instance, relation “friendship” connects two persons or relation “is reading” appears between a person and a book. Object originals are autonomic and they constantly evolve. The representation of relations between them is also subject to frequent changes. Even global information is highly evolutionary: changes on the object’s origin side (not in the representation in I) influence the semantics. That is, if an object corresponds to a database then updating its content can change the object’s relations to others. This type of dynamic semantics consolidation from the local semantics held by participating objects follows the emergent semantics approach for knowledge management [15]. The property corresponds to the P2P network topology maintenance problem.

Composition: The granularity level of objects provides an additional degree of freedom. One can consider a group of objects in I as a node in G_I a self-contained element with own semantic relations. For instance, a group of persons forms a team or a service is constructed as a chain of simpler services. From the P2P point of view, the composition property is similar to peer clustering and aggregation, including superpeer-based P2P systems.

Data integration: A smart space can be considered a virtual data integration system [17] for multiple sources. Some objects in I represent external data sources (e.g., databases) and the means to access data (or even reason knowledge over these data) from the sources. This property is conceptually close to hybrid P2P architectures and P2P-based search problem, including semantic-aware P2P systems.

Based on this P2P model we can translate some well-known P2P problems for use in smart spaces.

1. Nodes heterogeneity. Objects in I are of different concepts (even incomparable) of the application problem domain. It provides basic restrictions on node linkage in G_I. For instance, some nodes cannot be connected with a direct link or cannot be clustered together, similarly as it happens in structured P2P networks. The same restrictions exist in practical deployments of P2P systems due to the Internet Protocol (IP) level reachability factors (e.g., a NAT prevents

 ISBN: 978-1-61208-353-7

establishing a direct IP connection between two P2P nodes).

2. Neighbor selection. Every knowledge fragment should serve the system goals. It means that any object of I relates some other objects to form local semantics (over the data attributes the object has). When an object has many relations the knowledge becomes less concretized, thus, similarly to the P2P case, a node in G_I preferably keeps a moderate number of direct links. In P2P networks, a node has short-range and long-range links: the former is for nearby nodes, the latter allows jumping to distant area of the network. A short-range link in G_I describes a kind of persistent or system-wide knowledge. Similarly, emergent semantics provide long-range links for G_I, representing less stable knowledge relations.

3. Network topology maintenance. Objects can apply certain system-level rules when selecting neighbors, as it happens in structured P2P networks. The aim is at maintaining knowledge representation that allows efficient knowledge reasoning over I based on existing technologies of Semantic Web (e.g., SPARQL). To some extent, the maintenance can also preserve the consistency of collected knowledge.

4. Routing. Knowledge reasoning over I is reduced to traversal in G_I, when semantic relations between objects allows interpreting and then forming derivate knowledge. Knowledge can be defined as a connected subgraph in G_I. In particular, such a subgraph consists of a node and some paths starting from this node. Routing algorithms provide a way to construct such graphs. An object (node in G_I) acts as a client when it needs knowledge, a server when it completes knowledge reasoning, and a router when it forwards the construction to subsequent objects for additional knowledge.

In summary, the model allows considering content I as interacting objects, which are active entities (make actions) on one hand and are subject to information changes (actions consequence) on the other hand. Result of interaction is derived knowledge in a graph-based form. This fact allows us to describe formally the conceptual processes of service construction and delivery.

V. Service Construction and Delivery

From the information-centric point of view, we can consider a service as knowledge reasoning over the content I and delivering the result to the users. Conceptual steps of the service construction are algorithmically formalized in Figures 1 and 2 (adapted from [20]). Let o be a particular ontology used by the service. Write $[q(o) \to I]$ to denote the action of content retrieval. The result is either existential (yes/no) or constructive (found piece of information). Write $I + y$ and $I - y$ to denote the insertion and removal of information piece y, respectively.

The algorithm in Figure 1 embodies actions an information service. Step 1 detects when the service is needed based on the current situation in the smart space. Step 2 makes selection of knowledge x to deliver to the user. Step 3 decides which UI elements are target devices. Step 4 updates recent visualization v_d to include x on device d.

Algorithm: Information Service

Require: Ontology o to access information content I of the smart space. The set U of available UI devices.
1: Await $[q_{\mathrm{act}}(o) \to I] = \mathrm{true}$ {event-based activation}
2: Query $x := [q_{\mathrm{info}}(o) \to I]$ {information selection}
3: Select $d \in U$ {target UI devices}
4: Visualization $v_d := v_d + x$ {service delivery}

Figure 1. Actions in information service delivery.

Algorithm: Control Service

Require: Ontology o to access smart space information content I. The set U of available UI devices.
1: Await $[q_{\mathrm{act}}(o) \to I] = \mathrm{true}$ {event-based activation}
2: Query $x := [q_{\mathrm{info}}(o) \to I]$ {information selection}
3: Decide $y := f(x, o)$ {formulation of control action}
4: Update $I := I + y$ {service delivery}

Figure 2. Actions in control service delivery.

The algorithm in Figure 2 embodies actions of a control service. Step 1 analyzes the space content to detect that a control action is needed. Steps 2 and 3 are reasoning in context of the current situation, and the service decides what updates (possibly without human intervention) are needed in the recent system state. The updates become available to the participants (original in the physical and information worlds).

From the architectural point of view, a service is made by interaction of software agents, when each agent makes its contribution by changing objects in I. Moreover, an agent can be represented as an object $i \in I$ itself, similarly as in [9].

Now let us formally define a smart space service as a step-wise branching process of changing objects in I. Information content of any object involved into this process can be used as service outcome to deliver to users. The definition captures the following properties, which we illustrate below using SmartRoom system [20][21] in the examples.

1. Information service. The simplest case of a service is reading the information content of a given object representation in I. The only step of the process is that someone has published or updated the object.

Example: SmartRoom keeps (as objects in I) all human participants (person profiles) and their presentations. The latter has links to the files with slides, e.g., in PDF (Portable Document Format). Information on participants and presentations is accessed via the corresponding objects and then visualized on appropriate user interface (e.g., SmartRoom client that a participant runs on her/his mobile computer).

2. Control service. An informational service can be extended with control functions due to the virtualization property. If a controlled entity is represented as $i \in I$ then changing i leads to appropriate actions at the i's original.

Example: Presentation-service of SmartRoom follows up changes in the slide number of currently shown presentation. Whenever the number is updated the new slide is displayed on the SmartRoom wall screen (media projector).

3. Step-wise process. Smart spaces are event-based: a change of $i \in I$ forms an event observed by other participants. When $i_1 \in I$ is changed it can course creating or updating $i_2 \in I$, and so on. The process can be branched, i.e., one change affects many objects.

Example: When a new participant joins SmartRoom then several objects appear in I: person profile, presentation, time restrictions, etc. In turn, the activity agenda is updated (a speaker is added), adapting to the current situation.

Let us consider how the P2P model supports the structural description of virtualization and knowledge sharing in smart spaces. As a result, service construction can be formulated in terms of flows of information changes, which is convenient for the use in service design.

Given a starting object $s \in I$ and its initial change. Let $D(s)$ be a graph routable from s in G_I. Construction of a service corresponds to a routing path $s \to^* d$, as schematically depicted in Figure 3 using thick arrows. Injection of the change starts the service, analogous to a P2P node starting a lookup query. The sequence of changes flows in G_I. Note that parallel paths are possible. Any point when an agent reads an object can be considered a final step of the service construction since the agent consumes an outcome.

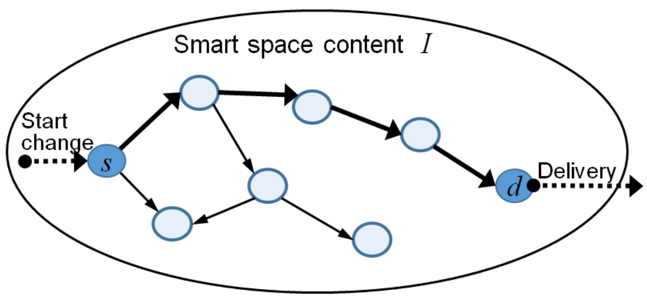

Figure 3. Service construction as P2P route $s \to^* d$.

This formalization is very flexible for various service constructions. There can be a large number of services due to freedom in selection of starting and final points.

Consider a path $s \to^* d$. There can be two types of links: ontological and mediatorial. An ontological link represents an object property from the ontology. Such a link is kept directly in I and is used in search queries of knowledge reasoning. A mediatorial link $i \to j$ is a result of actions of a participant (software agent): it analyzes object i and, as a consequence, changes another object j.

Figure 4 illustrates an example service "show me a slide". Thick arrows visualize the service construction process. Let user B browse available presentations kept in SmartRoom.

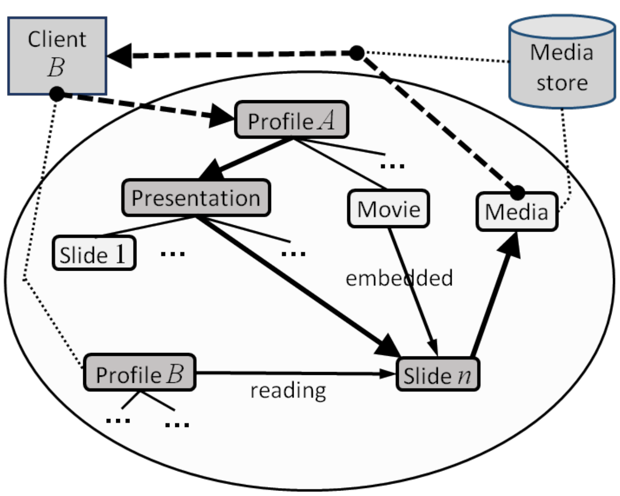

Figure 4. Example service construction for SmartRoom.

The B's client can find the A's presentation starting a path from Profile A and then running over hierarchical links till a given slide. The real slide (one-page PDF file) is physically located in SmartRoom media store (e.g., implemented as a web server). That is, Client B has to resolve the link from the smart space to the real content in the media store. Furthermore, if the recent slide embeds a movie, the latter is available for displaying by Client B from either the media store or the movie is located somewhere in public Internet and referenced by a global URL (Uniform Resource Locator, in particular use for web addressing).

As an additional effect of this service construction, more semantic relations can emerge in the smart space. In the example from Figure 4, the relation "reading" establishes the emergent semantics between B and the slide she or he is now analyzing.

VI. Conclusion

This paper addressed the problem of virtualization and knowledge sharing in smart spaces deployed in localized IoT environments. By this publication we want to initiate broad discussion on how to utilize well-known models for P2P networks for describing and making formal definition of virtualization and knowledge sharing in smart spaces. The paper summarized previous research on applicability of P2P methods for smart spaces. We proposed ideas on P2P modeling for virtualization and knowledge sharing in smart spaces. The discussion aims at the use of P2P models for service construction and delivery, following the M3 concept of smart spaces. We provided examples on how some well-known P2P problems, including P2P nodes heterogeneity, neighbor selection, network topology maintenance, and routing, are translated to smart spaces problems. Our P2P model allows structural description of virtualization and knowledge sharing, resulting in the definition of a smart space service in terms of information change flows. This description simplifies the problem of smart spaces design by providing enhanced abstractions for service construction and delivery.

Acknowledgment

This work is financially supported by Government of Russian Federation, Grant 074-U01. The research is a part of project 14.574.21.0060 (RFMEFI57414X0060) of Federal Target Program "Research and development on priority directions of scientific-technological complex of Russia for 2014–2020". The authors are grateful for DIGILE IoT SHOK program that provided required support of this research. The research of D. Korzun was funded by project # 1481 (basic part of state research assignment # 2014/154) of the Ministry of Education and Science of the Russian Federation.

References

[1] D. Miorandi, S. Sicari, F. De Pellegrini, and I. Chlamtac, "Survey internet of things: Vision, applications and research challenges," Ad Hoc Netw., vol. 10, no. 7, Sep. 2012, pp. 1497–1516.

[2] G. Kortuem, F. Kawsar, V. Sundramoorthy, and D. Fitton, "Smart objects as building blocks for the internet of things," IEEE Internet Computing, vol. 14, no. 1, Jan. 2010, pp. 44–51.

[3] D. J. Cook and S. K. Das, "Pervasive computing at scale: Transforming the state of the art," Pervasive Mob. Comput., vol. 8, no. 1, Feb. 2012, pp. 22–35.

[4] D. Korzun, S. Balandin, and A. Gurtov, "Deployment of Smart Spaces in Internet of Things: Overview of the design challenges," in Proc. 13th Int'l Conf. Next Generation Wired/Wireless Networking and 6th Conf. on Internet of Things and Smart Spaces (NEW2AN/ruSMART 2013), LNCS 8121, S. Balandin, S. Andreev, and Y. Koucheryavy, Eds. Springer-Verlag, Aug. 2013, pp. 48–59.

[5] E. Patouni, A. Merentitis, P. Panagiotopoulos, A. Glentis, and N. Alonistioti, "Network virtualisation trends: Virtually anything is possible by connecting the unconnected," in Proc. 2013 IEEE Software Defined Networks for Future Networks and Services (SDN4NFS). IEEE, Nov. 2013, pp. 1–7.

[6] S. Balandin and H. Waris, "Key properties in the development of smart spaces," in Proc. 5th Int'l Conf. Universal Access in Human-Computer Interaction (UAHCI '09). Part II: Intelligent and Ubiquitous Interaction Environments, LNCS 5615, C. Stephanidis, Ed. Springer-Verlag, 2009, pp. 3–12.

[7] E. Ovaska, T. S. Cinotti, and A. Toninelli, "The design principles and practices of interoperable smart spaces," in Advanced Design Approaches to Emerging Software Systems: Principles, Methodology and Tools, X. Liu and Y. Li, Eds. IGI Global, 2012, pp. 18–47.

[8] J. Kiljander, A. Ylisaukko-oja, J. Takalo-Mattila, M. Eteläperä, and J.-P. Soininen, "Enabling semantic technology empowered smart spaces," Journal of Computer Networks and Communications, vol. 2012, 2012, pp. 1–14.

[9] V. Gorodetsky, "Agents and distributed data mining in smart space: Challenges and perspectives," in Agents and Data Mining Interaction (ADMI 2012), LNAI 7607, L. Cao, Y. Zeng, A. Symeonidis, V. Gorodetsky, P. Yu, and M. Singh, Eds. Springer-Verlag, 2013, pp. 153–165.

[10] D. Korzun and A. Gurtov, Structured Peer-to-Peer Systems: Fundamentals of Hierarchical Organization, Routing, Scaling, and Security. Springer, 2013.

[11] S. Poslad, Ubiquitous Computing: Smart Devices, Environments and Interactions. John Wiley & Sons, Ltd, 2009.

[12] J. Honkola, H. Laine, R. Brown, and O. Tyrkkö, "Smart-M3 information sharing platform," in Proc. IEEE Symp. Computers and Communications (ISCC'10). IEEE Computer Society, Jun. 2010, pp. 1041–1046.

[13] D. Khushraj, O. Lassila, and T. W. Finin, "sTuples: Semantic tuple spaces," in Proc. 1st Annual Int'l Conf. Mobile and Ubiquitous Systems (MobiQuitous 2004). IEEE Computer Society, 2004, pp. 268–277.

[14] M. Palviainen and A. Katasonov, "Model and ontology-based development of smart space applications," Pervasive Computing and Communications Design and Deployment: Technologies, Trends, and Applications, May 2011, pp. 126–148.

[15] C. Aiello, T. Catarci, P. Ceravolo, E. Damiani, M. Scannapieco, and M. Viviani, "Emergent semantics in distributed knowledge management," in Evolution of the Web in Artificial Intelligence Environments, SCI 130, R. Nayak, N. Ichalkaranje, and L. Jain, Eds. Springer-Verlag, 2008, pp. 201–220.

[16] A. Y. Halevy and J. Madhavan, "Corpus-based knowledge representation," in Proc. 18th Int'l Joint Conf. on Artificial Intelligence (IJCAI'03). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2003, pp. 1567–1572.

[17] L. Bertossi and L. Bravo, "Consistent query answers in virtual data integration systems," in Inconsistency Tolerance, LNCS 3300, L. Bertossi, A. Hunter, and T. Schaub, Eds. Springer-Verlag, 2005, pp. 42–83.

[18] L. Pellegrino, F. Huet, F. Baude, and A. Alshabani, "A distributed publish/subscribe system for RDF data," in Data Management in Cloud, Grid and P2P Systems, LNCS 8059, A. Hameurlain, W. Rahayu, and D. Taniar, Eds. Springer-Verlag, 2013, pp. 39–50.

[19] M. Matuszewski and S. Balandin, "Peer-to-peer knowledge sharing in the mobile environment," in Proc. 5th Int'l Conf. on Creating, Connecting and Collaborating Through Computing (C5 '07). IEEE Computer Society, 2007, pp. 76–83.

[20] D. Korzun, I. Galov, A. Kashevnik, and S. Balandin, "Virtual shared workspace for smart spaces and M3-based case study," in Proc. 15th Conf. of Open Innovations Association FRUCT, S. Balandin and U. Trifonova, Eds. ITMO Univeristy, Apr. 2014, pp. 60–68.

[21] D. Korzun, I. Galov, and S. Balandin, "Smart room services on top of M3 spaces," in Proc. 14th Conf. of Open Innovations Association FRUCT, S. Balandin and U. Trifonova, Eds. SUAI, Nov. 2013, pp. 37–44.