Semantic Information Broker for Smart Spaces: Value Offering Deployment Options

Dmitry G. Korzun, Aleksandr A. Lomov, Ivan V. Galov

Petrozavodsk State University (PetrSU)

Petrozavodsk, Russia

{dkorzun, lomov, galov}@cs.karelia.ru

Abstract—A smart space is created by deploying a Semantic Information Broker (SIB) in order to enable information sharing for digital devices in their computing environment. In this work we discuss options for SIB deployment with the focus on the packaged software property. Two directions for SIB deployment options are studied: 1) deployment in dependence on a hosting computing environment for SIB and 2) augmented software deployed with SIB to perform specified knowledge processing. Our discussion moves the consideration of SIB from a pure programming result to a packaged software solution that being deployed can make value offering for the end-users.

Smart spaces form a programming paradigm for creating a certain class of ubiquitous computing environments [1]. Such an environment is associated with a physical spatial-restricted place (office, room, home, city square, etc.) equipped with a variety of devices (sensors, data processors, actuators, consumer electronics, personal mobile devices, multimodal systems, etc.) and, in addition to local networking has access to the Internet with its diversity of services and resources. A smart space enables information sharing in the environment, supporting construction of advanced digital services, which now are called smart.

The M3 architecture provides a particular approach to creation of smart spaces [2]. A smart space is created and maintained by deploying a Semantic Information Broker (SIB) on a host accessible by any device of the environment. SIB maintains an RDF triplestore forming a shared knowledge corpus for all participating devices. They run knowledge processors (KPs)—a specific form of software agents. Communication between them is indirect: KPs share information as RDF triples using the SIB for mediation. Created smart space is used to implement service-oriented applications. Each application is represented by a mush up of interacting KPs. They cooperatively construct services based on the provided mechanisms for semantic-oriented information sharing.

There are several SIB implementations available: the first official Smart-M3 SIB [3], RIBS [4] for resource limited devices, OSGi SIB [5] for Java-based systems, and RedSIB [6]. A modular SIB design was proposed in our previous work [7] to support such important properties as extensibility, dependability, and portability. The design is implemented in the CuteSIB release (distributed as open source software available at sourceforge.net/projects/smart-m3/).

In this work we discuss possible solutions for SIB deployment with the focus on the packaged software property. We consider such solutions as a step towards value offering for service providers and developers from the M3 based applications. Note that SIB is a technological solution that does not offer value in itself. Value can be created by deploying SIB and its augmented software in a given environment.

First, we consider SIB deployment in dependence on a hosting computing environment. From the value-offering point of view, some promising options are introduced in Table I. Importantly that SIB deployment is not limited by a certain class of computing environments, and a smart space can be created for any environment with a suitable host machine.

Second, we consider how to make a SIB a product with service value some augmented software should be deployed with the SIB. The following options can be exploited based on the fact that SIB is a middleware system.

1) Additional modules are built in SIB.
2) Additional KPs are packaged with SIB.

The first option explicitly advances the SIB with certain knowledge processing functions (e.g., in the form of plugins or added modules). For instance, the CuteSIB design [7] supports built-in reasoning (if needed the appropriate module is added at the SIB compilation phase). Deployed SIB analyzes the collected information and activity of KPs in order to deduce new knowledge. The latter is used by SIB itself for maintaining KPs operation or for enriching the shared knowledge corpus with the deduced knowledge. This kind of knowledge processing is important for system-level tasks such as access control [8], agent substitution [9], or subscription notification delivery control [10].

TABLE I.   SIB DEPLOYMENT OPTIONS

SIB host	Possible service-oriented applications
Local machine	Created smart space is used to implement a service-oriented information system accessed within the corporate network. Candidates for the local host machine are not limited with traditional servers: a desktop, laptop, or even single-board computer (e.g., Raspberry Pi) can be used.
WLAN router	A particular case of the local machine option. Created smart space is used to implement digital services for end-users located in the room or some indoor area. A similar option can be used for small-sized outdoor areas. Note that other non-traditional computers can host a SIB, e.g., a media-projector.
Personal mobile computer	Created smart space is used to implement mobile services that accompany the end-user, exploiting the capacity of her/his smartphone or tablet. The option advances the standalone mobile application style by making the mobile computer a small local server (which can access external resources from the global Internet).
Internet server	Created smart space is used to implement ubiquitously accesses services, advancing the web-server style of digital service provision. Although the option supports creating a smart space not associated with a physical spatial-restricted place, the localization is typically revealed due to the limited set of participants (end-users, devices, or external services).

In the second option, similar knowledge processing functions are delegated to the layers above the SIB layer, as shown in Fig. 1. The upper three layers are for KPs in dependence on their role in the applications. KPs on the g-KP layer perform processing useful for many services, even from different domains. Examples include local media-content sharing [11], local web-page construction [12], and context-based access control [13]. KPs on the d-KP layer produce information needed in a limited set of use cases within a certain problem domain. For instance, recommendations construction for e-Tourism domain [14] or monitoring of physiological activity of a mobile user [15]. KPs on the c-KP layer are responsible for service delivery to the end-users. For example, such a KP implements a personal mobile client on the smartphone [10].
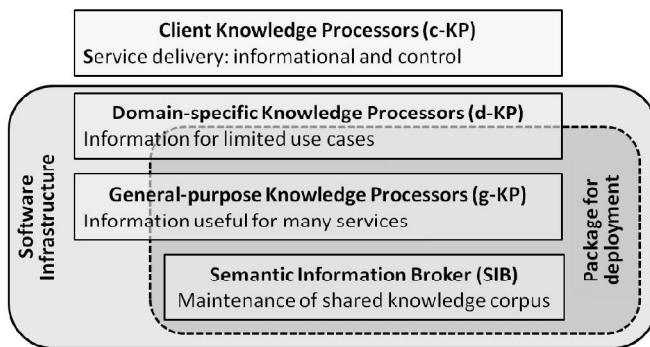

Fig. 1.   Service-valued package includes SIB and some infrastructural KPs

The notion of software infrastructure [11] for a smart space covers all the layers except the i-KP layer. The KPs for packaging with SIB represent a part of the infrastructure. They preferably run on the same host device with the SIB. As a result, the deployed system (SIB plus KPs) can construct services, in contrast to the option of SIB-only deployment. Service delivery to the end-users needs introduction of client KPs, which are not included into the package. Client KPs are installed and run on user interface devices by interested end-users. The packaged system also allows extension with additional infrastructural KPs from the d-KP and g-KP layers, which are not included into the package.

In summary, the above discussion showed how the existing consideration of SIB as a pure programming result can be evolved towards a packaged software solution. Such a solution, being deployed in a given computing environment, can already offer value for the end-users consuming services in this environment.

ACKNOWLEDGMENT

REFERENCES

[1]   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.

[2]   I. Oliver and J. Honkola, "Personal semantic web through a space based computing environment," *Computing Research Repository (CoRR)*, vol. abs/0808.1455, Aug. 2008.

[3]   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.

[4]   J. Suomalainen, P. Hyttinen, and P. Tarvainen, "Secure information sharing between heterogeneous embedded devices," in *Proc. 4th European Conf. Software Architecture (ECSA '10): Companion Volume*.   ACM, 2010, pp. 205–212.

[5]   D. Manzaroli, L. Roffia, T. S. Cinotti, E. Ovaska, P. Azzoni, V. Nannini, and S. Mattarozzi, "Smart-M3 and OSGi: The interoperability platform," in *Proc. IEEE Symp. Computers and Communications (ISCC'10)*.   IEEE Computer Society, 2010, pp. 1053–1058.

[6]   F. Morandi, L. Roffia, A. D'Elia, F. Vergari, and T. S. Cinotti, "RedSib: a Smart-M3 semantic information broker implementation," in *Proc. 12th Conf. of Open Innovations Association FRUCT and Seminar on e-Tourism*, S. Balandin and A. Ovchinnikov, Eds.   SUAI, Nov. 2012, pp. 86–98.

[7]   I. Galov, A. Lomov, and D. Korzun, "Design of semantic information broker for localized computing environments in the Internet of Things," in *Proc. 17th Conf. of Open Innovations Association FRUCT*.   ITMO Univeristy, IEEE, Apr. 2015, pp. 36–43.

[8]   A. D'Elia, D. Manzaroli, J. Honkola, and T. S. Cinotti, "Access control at triple level: Specification and enforcement of a simple RDF model to support concurrent applications in smart environments," in *Proc. 11th Int'l Conf. Next Generation Wired/Wireless Networking (NEW2AN'11) and 4th Conf. Smart Spaces (ruSMART'11)*.   Springer-Verlag, 2011, pp. 63–74.

[9]   I. Paramonov, A. Vasilev, D. Laure, and I. Timofeev, "Agent substitution mechanism for dataflow networks: Case study and implementation in Smart-M3," in *Proc. 13th Int'l Conf. Next Generation Wired/Wireless Networking and 6th Conf. on Internet of Things and Smart Spaces (NEW2AN/ruSMART 2013)*, ser. LNCS 8121, S. Balandin, S. Andreev, and Y. Koucheryavy, Eds.   Springer-Verlag, Aug. 2013, pp. 60–71.

[10]   D. Korzun, M. Pagano, and A. Vdovenko, "Control strategies of subscription notification delivery in smart spaces," in *Distributed computer and communication networks. CCIS 601*, V. Vishnevsky and D. Kozyrev, Eds.   Springer International Publishing, 2016.

[11]   I. Galov and D. Korzun, "The SmartRoom infrastructure: Service runtime reliability," in *Proc. 14th Conf. of Open Innovations Association FRUCT*, S. Balandin and U. Trifonova, Eds.   SUAI, Nov. 2013, pp. 188–189.

[12]   A. Vdovenko, S. Marchenkov, and D. Korzun, "Enhancing the smartroom system with e-tourism services," in *Proc. 17th Conf. Open Innovations Framework Program FRUCT*, Apr. 2015, pp. 237–246.

[13]   A. Smirnov, A. Kashevnik, N. Shilov, and N. Teslya, "Context-based access control model for smart space," in *Proc. 5th Int'l Conf. on Cyber Conflict (CyCon 2013)*, Jun. 2013, pp. 1–15.

[14]   A. Smirnov, A. Kashevnik, A. Ponomarev, N. Teslya, M. Shchekotov, and S. Balandin, "Smart space-based tourist recommendation system," in *Proc. 14th Int'l Conf. Next Generation Wired/Wireless Networking and 7th Conf. on Internet of Things and Smart Spaces (NEW2AN/ruSMART 2014), LNCS 8638*, S. Balandin, S. Andreev, and Y. Koucheryavy, Eds.   Springer, Aug. 2014, pp. 40–51.

[15]   D. Korzun, A. Borodin, I. Timofeev, I. Paramonov, and S. Balandin, "Digital assistance services for emergency situations in personalized mobile healthcare: Smart space based approach," in *Proc. 2015 Int'l Conf. on Biomedical Engineering and Computational Technologies (SIBIRCON/SibMedInfo)*.   IEEE, Oct. 2015, pp. 62–67.