Performance Evaluation of Smart-M3 Applications: A SmartRoom Case Study

Dmitry G. Korzun, Sergey A. Marchenkov,
Andrey S. Vdovenko, Andrey N. Borodulin
Petrozavodsk State University (PetrSU)
Petrozavodsk, Russia
{dkorzun, marchenk, vdovenko, boroduli}@cs.karelia.ru

Sergey I. Balandin
FRUCT Oy
Helsinki, Finland;
ITMO University
Saint-Petersburg, Russia
Sergey.Balandin@fruct.org

Abstract—**Smart spaces support development of advanced service-oriented applications that introduce intelligence into Internet of Things environments. Current development meets the performance challenge, since the intelligence is achieved by the cost of performance. In this paper, we experimentally study a representative application for smart spaces—the SmartRoom system. It assists humans in such collaborative work activity as conferences, meetings, and seminars. Using the existing software Smart-M3-based implementation, we evaluate the performance of the three important components of service construction and delivery within a localized IoT environment of the room: 1) construction of the core SmartRoom services, 2) access of personal mobile devices to the system for delivering services, and 3) use of services composed with external Internet services. Our experiments confirm the Smart-M3 applicability. Even in the settings of a restricted-capacity IoT environment, the performance is suitable for practical purpose. The evaluated system components are common, and our experimental conclusions provide performance bounds convenient for a wide class of Smart-M3 applications.**

I. INTRODUCTION

Smart spaces support development of service-oriented applications that introduce intelligence into Internet of Things (IoT) environments [1], [2], [3]. A lot of work exists on "smart" applications, see [4], [5], [6], [7] and references therein. A smart space can be thought as an advanced computing environment that acquire and apply knowledge to adapt services in order to enhance user experience [8], [7]. A smart space participant is an autonomous information processing unit that runs on a device of the environment. Services are constructed as a result of interaction over shared information. This notion of smart space was adopted by the M3 architecture and implemented in Smart-M3 platform [9], [10], [11].

The Internet of Things (IoT) supports ubiquitous connectivity property for smart spaces. We consider smart spaces deployed in localized IoT environments; each is typically associated with a physical spatial-restricted place (office, room, home, city square, etc.). The environment is equipped with variety of devices: sensors, data processors, actuators, consumer electronics, personal mobile devices, multimodal systems, etc. Many IoT devices become operating in the environment and realizing continued processing of many data flows, originated from various sources and consumed by multiple applications.

For such IoT environments, digital services of the next generation (smart applications) are developed and deployed. The physical world becomes digitalized and connected with the information world. This phenomenon refers to the term of smart environment, forming a new challenging direction of networked software engineering. The intelligence that a smart environment introduces into its services is achieved by the cost of the system complexity. When many heterogeneous participants are involved into the distributed system and the IoT environment is subject to asynchronous dynamic changes and operation uncertainty then the system performance clearly becomes an issue.

The scope of this paper is limited with the M3 architecture (multi-device, multi-vendor, multi-domain) for smart spaces [9], [12], [13]. It utilizes the blackboard and publish/subscribe (Pub/Sub) architectural patterns to share information in the environment, rather than have the devices explicitly send messages to one another. The information and its semantics are collected in a smart space using ontological representation models of the Semantic Web and forming a knowledge base for interoperable information sharing among participants. In particular, the M3 architecture is implemented in Smart-M3 platform, which provides open source middleware for development and deployment of smart spaces in various IoT environments (sourceforge.net/projects/smart-m3).

The current smart space-based applications development for localized IoT environments meets the performance challenge, since the intelligence is achieved by the cost of performance. In this paper, we experimentally study the SmartRoom system [14], [7], which assists humans in such collaborative work activity as conferences, meetings, and seminars. The system can be treated as a representative for a wide class of smart space-based applications, and its evaluation conclusions are applicable as a reference case.

The existing software implementation of SmartRoom is based on the Smart-M3 platform and available as open source (sourceforge.net/projects/smartroom). The system is deployed within the localized IoT environment of a multimedia equipped room. We evaluate the performance of the three important SmartRoom components for service construction and delivery:

1) local construction of the core SmartRoom services,
2) access of personal mobile devices to the system,
3) use of services composed with external resources.

Our experimental study confirms that even in a restricted-capacity localized IoT environment, the Smart-M3 platforms ensures the performance suitable for the considered class of

ISSN 2305-7254

smart space-based applications. Since the evaluated system components are common, our experimental conclusions show performance bounds that are convenient for many Smart-M3 applications.

The rest of the paper is organized as follows. Section II overviews the key aspects of smart space-based application development. Section III introduces the SmartRoom system as a representative smart space-based application. The subsequent sections consider experimental performance evaluation. Section IV discusses our experimental results for the core services construction in a smart space. Section V discusses our experimental results for the wireless network access to the system. Section VI discusses our experimental results for the use of services composed with external services from the Internet. Section VII concludes the paper.

II. RELATED WORK

Oliver [15] introduced the notion of smart space (information space) as being maintained by a local information broker. Interaction with the space is indirect, via agents (residing on suitable devices). They contact the broker for data sharing in the space. Data from the surrounding environment and global information of the Web as well as semantics, reasoning and processing about this information can be localized and personalized within such spaces. This notion of smart space then was adopted by the M3 architecture and implemented in Smart-M3 platform [9]. The latest open source implementations are RedSIB [10] and CuteSIB [11].

The original Smart-M3 work of Honkola *et al.* [9] considered several case studies of smart space-based applications with no experimental evaluation. Several measurement studies, including [16], [10], [17], [6], evaluated the performance of basic Smart-M3 read&write operations for agents to share information in the smart space. Bhardwaj *et al.* [18] analyzed the end-to-end delay between interacting agents. Niezen *et al.* [19] evaluated the performance of query resolution and reasoning mechanisms for agents resided on embedded and consumer electronics devices. Vanag and Korzun [20] analyzed the performance of resolving advanced search queries. Galov and Korzun [21] and Korzun *et al.* [22] evaluated the efficiency of notification delivery to the agents subscribed to information in a smart space. Paramonov *et al.* [23] considered the mechanism for a failed agent to be substituted with another one. Galov and Korzun [24] estimated the performance for an agent to restart in or reconnect to the smart space.

The above experimental research analyzed the basic Smart-M3 platform mechanisms for agents to interact by information exchange via a local broker. In contrast, this paper considers the Smart-M3 application capability to service construction and delivery to the users. The experimental evaluation uses the SmartRoom system. The system is a representative of a wide class of smart space-based applications with local service construction in the IoT environment, with mobile users as primary service consumers, and with services composed with external resources.

III. SMARTROOM SYSTEM

Today's personal mobile devices, such as smartphones and tablet computers, can be used to support people in effective communicating and working together in a multimedia equipped room with network access to the global Internet. The collective solving process is composed from provisions of the participants. Personal mobile devices support human participation in the process, when any participant can access digital assistance services directly from the personal device. The SmartRoom system is a representative smart space based application for this emerging IoT application domain of collaborative work, see [14], [7] and references therein. The high-level SmartRoom architecture is depicted in Fig. 1.

To create a smart space for SmartRoom we employ the Smart-M3 platform [9], [10], [11]. It is developed my international community and distributed as open source (http://sourceforge.net/projects/smart-m3/). The key architectural component is Semantic Information Broker (SIB) that implements an information hub for agents of a given IoT environment. Agents are also called knowledge processors (KPs). They run on devices of the environment. Some of them act on behalf of external data sources, resources, and services.

The IoT environment is localized in a physical spatial-restricted room equipped with various computing (e.g. server machines, computers) and media devices (e.g. projectors, TVs, interactive boards). Devices are connected via wireless and wired local area networks. Software infrastructure of SmartRoom provides means for application operation and ensures the application is operating in a proper way. The infrastructure includes: a) system software, which enables operation and network communication, b) the SIB, and c) KPs responsible for core service construction and delivery. The server machine running the SIB is either physically presented in the room or remote (e.g., in a corporate network). Computers running KPs for the core services are usually presented in the room.

The core SmartRoom services are Presentation-service and Conference-service. They provide the basic functionality

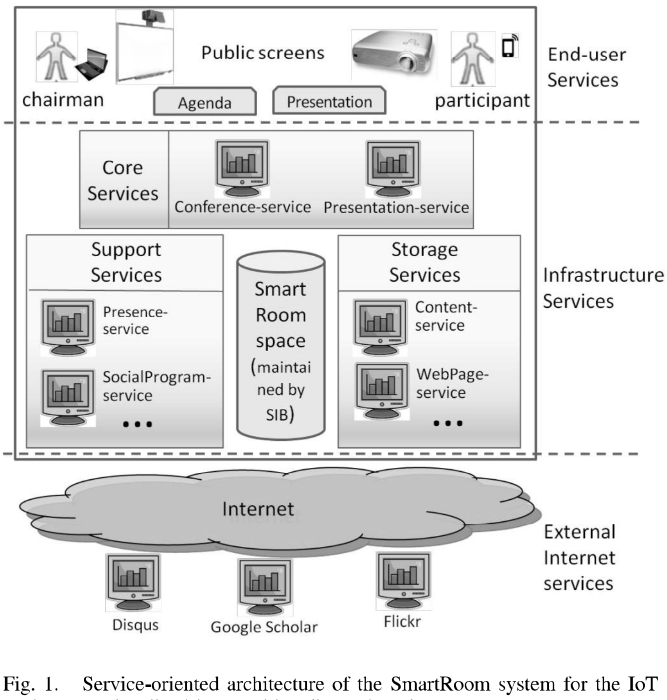

Fig. 1. Service-oriented architecture of the SmartRoom system for the IoT environment localized in a multimedia equipped room

associated with automated holding of collaborative activities such as conferences. Presentation-service displays multimedia presentations and operates with the related content shared over the SmartRoom space (a link to access the presentation or video, the total number of slides, the number of current slide). Presentation-service is constructed by a single KP running on a local computer connected to the projector. Conference-service dynamically maintains the activity program (i.e., conference section or agenda of talks).

In addition to public screens, each SmartRoom participant can use own personal mobile device, such as smartphone or tablet [14]. Being in the room and with the device connected to the wireless local area network (WLAN), the participant runs a client (SR-client) on her/his device. Let us consider such a participant an explicit SmartRoom user, since she/he can directly access the system and consume services using the client. The client becomes aware of currently available services and enables the user to select and access (in a personalized manner) a subset of the desired services from the total set of SmartRoom services for all participants. In particular, browsing the information on her/his personal device, the user can access Agenda-service and Presentation-service with no need to watch the public screens.

An example of a SmartRoom service composed with an external Internet service is Discussion-service. This SmartRoom service allows the participants to discuss by publishing commentaries on conference talks, either currently ongoing, or already passed ones, or just going to happen. The Discussion-service KP fetches from the SmartRoom space the total number of talks within the section, along with their titles/topics and names of presenters. Based on this information, the KP makes queries to the external Internet service—blog comment hosting service Disqus for web sites and online communities (https://disqus.com/). The external service replies with links to widgets, which enable the discussion feature. Then a web page is created for each talk, embedding the discussion widget into the page. This way, SmartRoom participants join a discussion thread of the talk by browsing its web page (either from SmartRoom clients or by web browsers from any computer). The web pages are regularly updated based on the latest participants' activity. For easy navigation, a summary web page is created to list all available discussion threads. Moreover, it provides auto-navigate function, i.e., when the user accesses the service during certain talk, she/he will be redirected to the page with the discussion widget of that talk.

Characteristics of a possible IoT environment for the SmartRoom system are listed in Table III. We intentionally consider a restricted-capacity environment in the performance

TABLE I. TESTBED FOR EXPERIMENTS

SIB server machine	CPU Intel Xeon, CPU E5-2630 2.30 GHz, RAM 4 Gb, HDD 36 Gb
Local computers with Wi-Fi access	CPU Intel Core, CPU 2.30–2.50 GHz, RAM 8Gb
WLAN in the testbed room	CPU Broadcom BCM5300, RAM 256Mb, WLAN-adapter: Broadcom BCM43xx 802.11b/g/n, Internet access, download speed from SIB server: 20.85 Mbps, DD-WRT firmware
Internet access from WLAN to SIB server	Download speed from SIB server: 20.85 Mbps, RTT to SIB server: 4.5 ms
Internet access from WLAN to external Internet services	Download speed from remote Internet servers: 0.81 Mbps, RTT to external servers: 12 ms

terms of WLAN, Internet access, and local computers. In the subsequent sections, this environment is exploited as a testbed to evaluate practical performance bounds. Note that recent technology progress allows creating relatively cheap environments of higher capacity. WLAN is specifically set to the low bandwidth for all local mobile devices. Local computers are typical laptops with Windows and Linux operating systems. Personal mobile devices are simulated on a single local computer due to the need of scalability evaluation with a few dozen or hundred human participants.

IV. CORE SMARTROOM SERVICES

The most processing part of core SmartRoom services construction is performed locally, on computers installed in the room. In particular, the computer of Presentation-service KP is connected with the public screen to display media information during talks (e.g., slide show). The basic scenario scheme is shown in Fig. 2.

Presentation-service is constructed based on Content-service [24]. The latter is a SmartRoom storage service to keep such multimedia content as files of presentations, see Fig. 1 above. Each PDF file of a presentation is sliced into separate PDF files, one per slide. The Content-service KP acts as a web server accepting download requests for the stored files. The link to every file is shared in the SmartRoom space, making the files available to SmartRoom KPs.

Conference-service shares appropriate links to the files with slides in the smart space. Presentation-service is responsible for visualization of a current speaker's slide show (her/his recent slide of the whole presentation) on the public screen. Presentation-service keeps information on the number of current slide and the total count of slides in the presentation. This solution allows the service to download the current slide using a web link to the file and then to show the slide on the public screen. The considered operations introduce small load

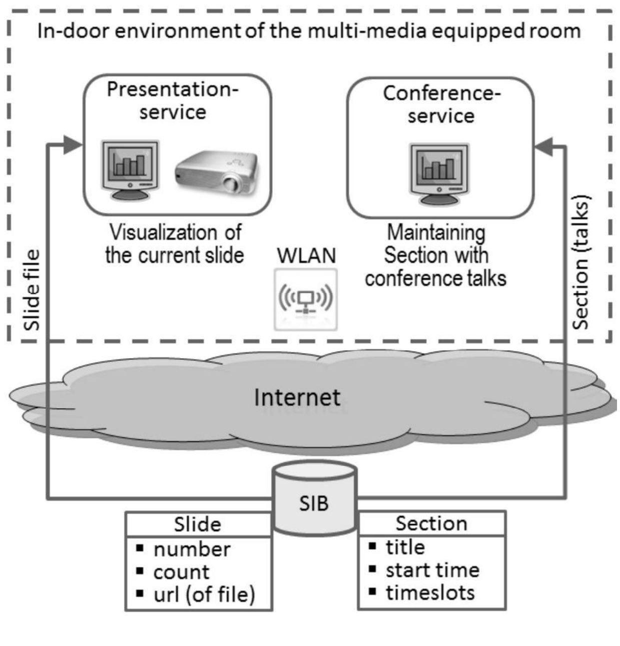

Fig. 2. Core SmartRoom services: local service construction

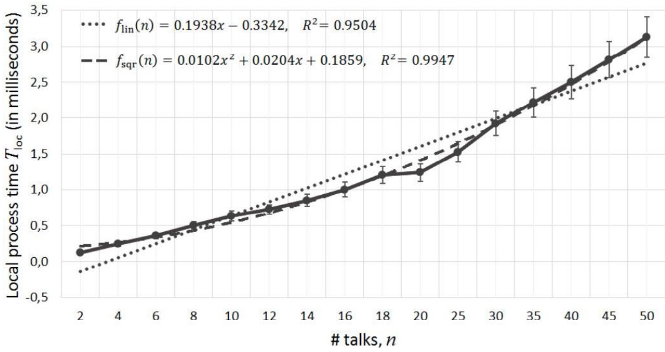

Fig. 3. Estimated average local time $T_{\text{loc}}(n)$ of processing n-talks section with the core SmartRoom service

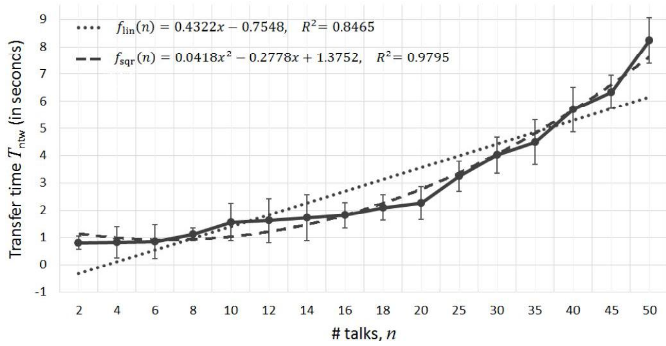

Fig. 4. Estimated average time $T_{\text{ntw}}(n)$ of transferring n-talks section from SIB to the core SmartRoom service

to Presentation-service; the load depends only on the file size of the slide (from 50 KB to 250 KB).

Each conference section consists of talks to be presented by the conference participants during the collaborative activity in the room. Full description of a conference section is shared in the smart space. Consequently, one of the resource-expensive operations is for a service to read a section from SIB by network communication (time T_{ntw}) and to process the section locally (time T_{loc}). The service performs these two operation steps whenever the section is updated in SIB, e.g., a new speaker is added or timeslot of the talk is adjusted. Maintaining section is the service process of transformation from an ontological representation format to the local data structures used in programming language. This allows the service to store the information of talks locally and dynamically maintain the activity program using only subscription to the change notification.

Each section is described in an ontological representation format (to be shared in the smart space). The representation includes an OWL individual (instance of ontology class) with properties describing each talk (title, start time, time slots). For n talks, the number of properties equals to $3 + 4n$. For example, for $n = 10$ such an OWL individual needs about 1 Kb of network transfer (an XML file of RDF triples).

Experimental behavior of $T_{\text{loc}}(n)$ and $T_{\text{ntw}}(n)$ is shown in Fig. 3 and Fig. 4, respectively. The number of talks is varied for $2 \leq n \leq 50$ to analyze the scalability for variety formats of section. For each n we make 100 runs, and the plot shows the average behavior with standard deviation bars. The behavior of $T_{\text{loc}}(n)$ is close to linear. The growth of $T_{\text{ntw}}(n)$ is faster than linear and closer to polynomial. The presented plots include the linear and the exponential regression.

We observe that for typical-size sections (8–16 talks) the sum of the local process and the transfer times does not exceed 2 s, which is satisfactory for this class of applications. However, 8 s to process and transfer 50 talks might be too much time, especially taking into account the non-linear growth of $T_{\text{ntw}}(n)$. The standard deviation of the transfer time is 0.58 s on average, i.e., the network communication is highly varied due to other workload. Consequently, if the application needs higher performance then the network capacity is increased while the SIB capacity is preserved.

In the recent implementation, when a section is updated, the service retrieves from the SIB the full RDF tree representing the talks. We plan to utilize the SPARQL query language and retrieve only the updated parts of the section. This way will certainly decrease the average time T_{ntw}. Note that we experimented in a restricted-capacity wireless environment (see Table III), thus the presented estimates provide worst-case upper bounds.

V. Mobile access to SmartRoom services

SmartRoom participants use their personal mobile devices to access services via SmartRoom clients [14]. In particular, each participant can view the current presentation as a slide show on own device (e.g., smartphone or laptop), in addition to the public screen of Presentation-service. The scheme for this scenario is shown in Fig. 5.

Implementation of this scenario utilizes the Smart-M3 subscription operation: SmartRoom SIB notifies every client (that follow slide changing) whenever the current slide number is

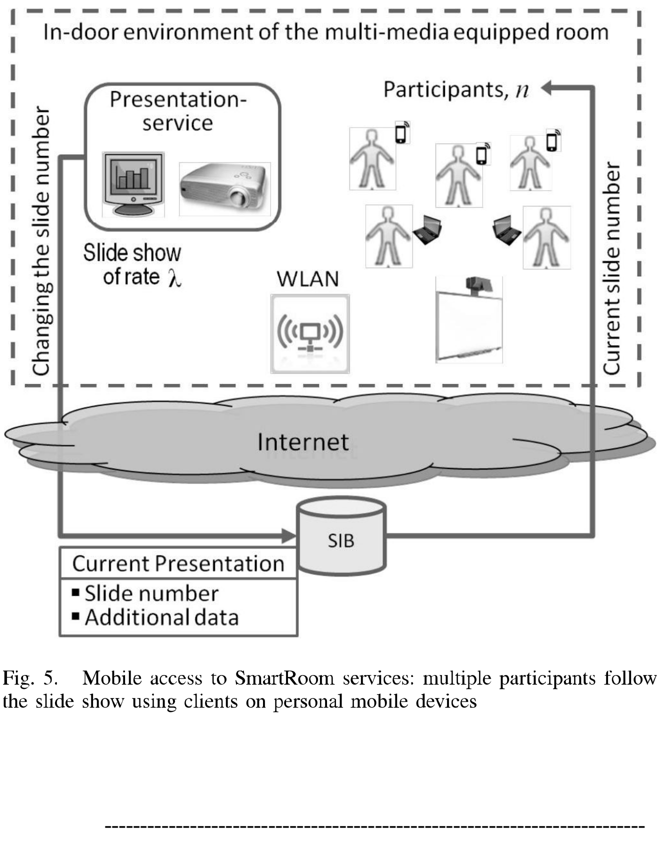

Fig. 5. Mobile access to SmartRoom services: multiple participants follow the slide show using clients on personal mobile devices

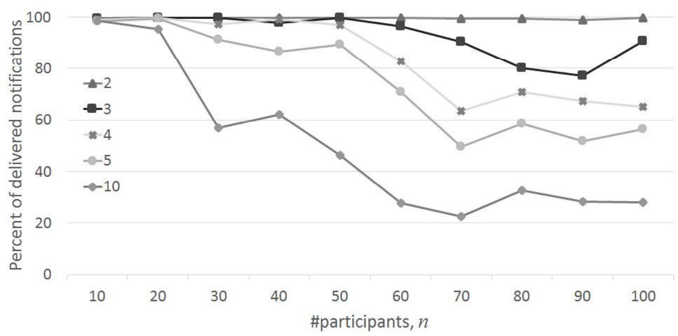

Fig. 6. Estimated percent of successful notification delivery to multiple clients

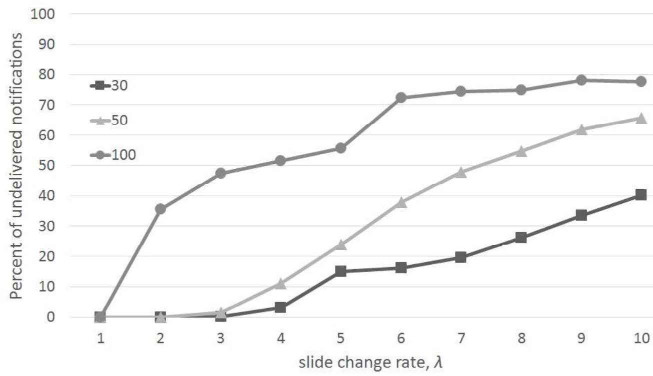

Fig. 7. Estimated percent of lost notifications

changed. Clearly, the key performance factors are the number of clients n and the rate λ of slide changing (initiated at Presentation-service). Previous work [10], [25], [22] observed that the subscription operation in smart spaces is subject to performance degradation due to losses.

We simulate operation of multiple mobile clients: all $0 < n < 100$ clients run on a single local computer in the SmartRoom WLAN (see Table III). Each client independently establishes one subscription to follow current slide changing. For given n, Presentation-service publishes $m = 150$ slide changes with fixed rate λ. In total, $N = nm$ notifications for given n and λ are expected to be delivered to the clients. Different values $1 \leq \lambda \leq 10$ s^{-1} are used in the experiments (i.e., one slide change happens on interval of length μ, where $1 \geq \mu = 1/\lambda \geq 0.1$ s). The reported experiments used CuteSIB [11]—the latest implementation variant for Smart-M3 platform [9].

Let $\sigma = \sigma(n;\lambda)$ be the share of successfully delivered notifications for all n clients. Any client is subscribed to the same information fragment, which is changed with rate λ. Then the relative subscription notification loss is $1 - \sigma(n;\lambda)$. Intuitively, the function $\sigma(n;\lambda)$ is decreasing. Clearly, growth of the number of clients n or the slide change rate λ reduces the share of successfully delivered notifications.

Experimental behavior of $\sigma(n;\lambda)$ is shown (on average, in percents) in Fig. 6. The number of clients n is varied and each experimental curve is shown for a fixed value of the change rates λ. The presented stress case of high values for λ shows the scalability limits of the recent SIB implementation. More than half of notifications can be lost when changes are too frequent and there many subscribers. Experiment results for smaller rates $\lambda < 2$ and with $n \leq 10$ are not presented in the plot, since for this moderate case no notification losses are observed or they are negligible.

Experimental behavior $1 - \sigma(n;\lambda)$ is shown (on average, in percents) in Fig. 7. The change rate λ is varied and experimental curves are shown for fixed $n = 30, 50, 100$. For the high rate λ the notification loss becomes unacceptable. A possible solution is increasing the capacity of SIB (server machine) and of the Wi-Fi network.

The analysis of centrality and variability of the subscription notification loss $1-\sigma(n;\lambda)$ is presented in Table II. In addition to the average σ_{avg} (also plotted in relative values in Fig. 7) and standard deviation σ_{sdv} we estimated percentiles Q_p for several $0 < p < 100\%$ The average σ_{avg} and median $\sigma_{\text{med}} = Q_{50}$ are observed to be close. The percentiles Q_{90} and Q_{10} show the upper and lower bounds for 80% of the observed losses. Half of the observed losses differ at most on $Q_{75} - Q_{25}$.

Therefore, we can conclude that the SIB capacity is enough for the considered class of applications: there are several dozen participants and any slide typically remains unchanged several seconds or even minutes.

TABLE II. ESTIMATED CENTRALITY AND VARIABILITY MEASURES FOR THE SHARE OF LOST NOTIFICATIONS

n	λ	1	2	3	4	5	6	7	8	9	10
30	Q_{90}	0	0	0	6	18	25	26	29	38	46
	$\sigma_{\text{med}} = Q_{50}$	0	0	0	2	14	16	24	26	34	40
	σ_{avg}	0	0	0	3	15	16	19	26	34	40
	Q_{10}	0	0	0	1	13	4	5	23	31	35
	σ_{sdv}	0	0	0	2	2	8	9	3	3	6
	Q_{75} - Q_{25}	0	0	0	4	3	8	12	5	3	3
50	Q_{90}	0	0	3	14	29	41	56	58	69	70
	$\sigma_{\text{med}} = Q_{50}$	0	0	1	10	25	39	47	54	62	67
	σ_{avg}	0	0	1	11	24	38	48	55	62	66
	Q_{10}	0	0	0	9	17	31	41	52	58	58
	σ_{sdv}	0	0	2	2	5	4	7	6	6	5
	$Q_{75} - Q_{25}$	0	0	2	3	7	4	11	2	4	3
100	Q_{90}	0	39	51	53	69	79	80	82	86	83
	$\sigma_{\text{med}} = Q_{50}$	0	34	46	51	50	71	76	73	82	79
	σ_{avg}	0	36	47	51	56	72	74	75	78	78
	Q_{10}	0	31	45	50	47	65	68	70	71	74
	σ_{sdv}	0	7	3	2	10	6	5	5	8	6
	Q_{75} - Q_{25}	0	3	5	1	18	10	8	7	10	5

VI. USE OF EXTERNAL SERVICES

Discussion-service provides an example of using external Internet services in the SmartRoom sysetm. The recent implementation supports such an external service as Disqus (http://disqus.com/)—a blog comment hosting service for web sites. The basic scenario scheme of the composed service is shown in Fig. 8.

The local part of Discussion-service consists of modules Conference-handler and Chat. The former watches the conference section shared in the smart space and reacts whenever the SIB notifies on a change in the agenda. The latter is responsible for creating locally a web page that represents a current discussion state to visually deliver to interested participants. The discussion is actually formed at the external service. For web page creation the Chat module requests widgets from the

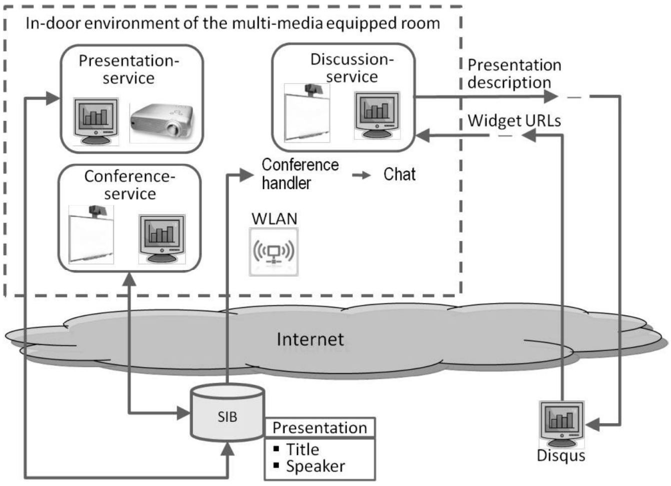

Fig. 8. Use of external Internet services in SmartRoom: a web page is created to integrate external widgets and local information

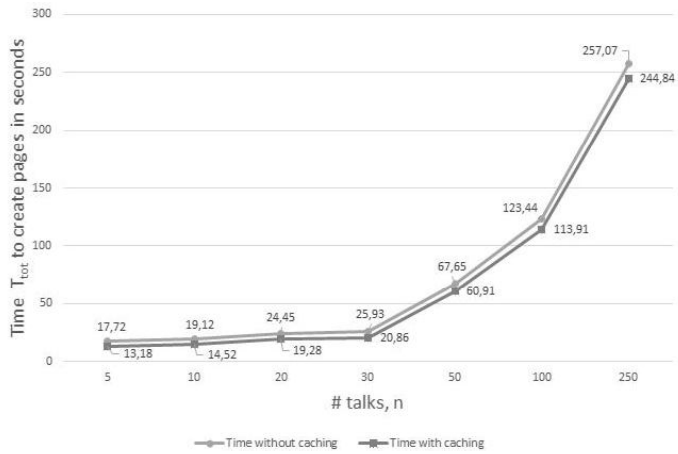

Fig. 9. Estimated time $T_{\rm tot}(n)$ spent for creating the web page

Disqus service. The number of widgets is equal to the number of talks n in the given conference section.

Let $T_{\rm wgt}$ be the time elapsed at the Chat module from the point of sending the first request to the Disqus service for a widget till the point of receiving the last widget URL. When all requested widgets are received, the Conference-handler module spends time $T_{\rm wpg}$ to create the web page containing all discussion threads derived from the widgets.

The experiment results for estimating $T_{\rm tot} = T_{\rm wgt} + T_{\rm wpg}$ are shown in Fig. 9. The upper curve corresponds to the case without caching on the side of external Internet service. Typically, all requests except the first one are with caching.

When the number of talks is in a moderate range (a typical conference section includes $n \leq 30$ talks), the behavior is close to linear. In this case, the performance is satisfactory since the service construction is completed in a few seconds.

The stress case with $30 < n \leq 250$ shows the scalability level of service construction. In this case, we observe that the performance quickly degrades. Several minutes are needed to complete the web page creation for an extremely big section. Note that the observed variability of $T_{\rm tot}(n)$ for a given n is low: the standard deviation is within $0.5 \dots 1.0$ s for the most values of n.

As in the previous experiments (Sections IV and V), we used a wireless environment of restricted-capacity (see Table III). Nevertheless, the observed time expenses of several seconds is enough for the considered class of applications.

VII. CONCLUSION

This paper presented results of several experiments for performance evaluation of a smart space based application. For the target application the SmartRoom system is selected to represent a reference case for a wide class of smart space-based applications. We evaluated the performance of the three important components of service construction and delivery within a localized IoT environment of the multimedia equipped room: 1) construction of the core SmartRoom services, 2) access of personal mobile devices to the system for delivering services, and 3) use of services composed with external Internet services. Our study showed that even in the settings of a restricted-capacity IoT environment, the performance is suitable for practical purpose. We conclude that during the last years the Smart-M3 platform has archived acceptable capability for the IoT settings.

ACKNOWLEDGMENT

This research is financially supported by the Ministry of Education and Science of the Russian Federation: 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 to all anonymous reviewers from the Open Innovations Associations FRUCT for their valuable comments.

REFERENCES

[1] 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, pp. 44–51, Jan. 2010.

[2] J. Augusto, V. Callaghan, D. Cook, A. Kameas, and I. Satoh, "Intelligent environments: a manifesto," *Human-centric Computing and Information Sciences*, vol. 3, no. 1, 2013.

[3] E. Balandina, S. Balandin, Y. Koucheryavy, and D. Mouromtsev, "IoT use cases in healthcare and tourism," in *Proc. IEEE 17th Conf. on Business Informatics (CBI)*, vol. 2, Jul. 2015, pp. 37–44.

[4] 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.

[5] S. Pantsar-Syvaniemi, E. Ovaska, S. Ferrari, T. S. Cinotti, G. Zamagni, L. Roffia, S. Mattarozzi, and V. Nannini, "Case study: Context-aware supervision of a smart maintenance process," in *Proc. 2011 IEEE/IPSJ Int'l Symposium on Applications and the Internet (SAINT '11)*. IEEE Computer Society, 2011, pp. 309–314.

[6] J. Kiljander, A. D'Elia, F. Morandi, P. Hyttinen, J. Takalo-Mattila, A. Ylisaukko-oja, J.-P. Soininen, and T. S. Cinotti, "Semantic interoperability architecture for pervasive computing and Internet of Things," *IEEE Access*, vol. 2, pp. 856–874, Aug. 2014.

[7] D. Korzun, A. Kashevnik, S. Balandin, and A. Smirnov, "The Smart-M3 platform: Experience of smart space application development for Internet of Things," in *Internet of Things, Smart Spaces, and Next Generation Networks and Systems. Proc. 15th Int'l Conf. Next Generation Wired/Wireless Networking and 8th Conf. on Internet of Things and*

Smart Spaces (NEW2AN/ruSMART 2015), LNCS 9247, S. Balandin, S. Andreev, and Y. Koucheryavy, Eds. Springer, Aug. 2015, pp. 56–67.

[8] D. J. Cook and S. K. Das, "How smart are our environments? an updated look at the state of the art," *Pervasive and Mobile Computing*, vol. 3, no. 2, pp. 53–73, 2007.

[9] 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.

[10] 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.

[11] 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.

[12] A. Smirnov, A. Kashnevik, N. Shilov, I. Oliver, S. Balandin, and S. Boldyrev, "Anonymous agent coordination in smart spaces: State-of-the-art," in *Smart Spaces and Next Generation Wired/Wireless Networking. Proc. 9th Int'l Conf. NEW2AN'09 and 2nd Conf. on Smart Spaces ruSMART 2009. LNCS 5764*. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 42–51.

[13] 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, pp. 1–14, 2012.

[14] A. Vdovenko, S. Marchenkov, and D. Korzun, "Mobile multi-service smart room client: Initial study for multi-platform development," in *Proc. 13th Conf. of Open Innovations Association FRUCT and 2nd Seminar on e-Tourism for Karelia and Oulu Region*, S. Balandin and U. Trifonova, Eds. SUAI, Apr. 2013, pp. 143–152.

[15] I. Oliver, "Information spaces as a basis for personalising the semantic web," in *Proc. 11th Int'l Conf. Enterprise Information Systems (ICEIS 2009)*, May 2009, pp. 179–184.

[16] M. Etelapera, J. Kiljander, and K. Keinanen, "Feasibility evaluation of M3 smart space broker implementations," in *Proc. 2011 IEEE/IPSJ 11th Int'l Symp. on Applications and the Internet (SAINT)*. IEEE, Jul. 2011, pp. 292–296.

[17] 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.

[18] S. Bhardwaj, T. Ozcelebi, R. Verhoeven, and J. Lukkien, "Delay performance in a semantic interoperability architecture," in *Proc. 2011 IEEE/IPSJ 11th Int'l Symp. on Applications and the Internet (SAINT)*, Jul. 2011, pp. 280–285.

[19] G. Niezen, B. J. J. van der Vlist, S. Bhardwaj, and T. Ozcelebi, "Performance evaluation of a semantic smart space deployment," in *Proc. 2012 IEEE Int'l Conf. on Pervasive Computing and Communications Workshops (PERCOM Workshops)*. IEEE, Mar. 2012, pp. 835–841.

[20] P. Vanag and D. Korzun, "SmartSlog knowledge patterns: Initial experimental performance evaluation," in *Proc. 11th Conf. of Open Innovations Association FRUCT and Seminar on e-Tourism*, S. Balandin and A. Ovchinnikov, Eds. SUAI, Apr. 2012, pp. 176–180.

[21] I. Galov and D. Korzun, "A notification model for Smart-M3 applications," 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-Verlag, Aug. 2014, pp. 121–132.

[22] 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.

[23] 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.

[24] I. Galov and D. Korzun, "Fault tolerance support of Smart-M3 application on the software infrastructure level," in *Proc. 16th Conf. of Open Innovations Association FRUCT*. ITMO Univeristy, Oct. 2014, pp. 16–23.

[25] A. A. Lomov and D. G. Korzun, "Subscription operation in Smart-M3," in *Proc. 10th Conf. of Open Innovations Association FRUCT and 2nd Finnish–Russian Mobile Linux Summit*, S. Balandin and A. Ovchinnikov, Eds. SUAI, Nov. 2011, pp. 83–94.