Delay performance in a semantic interoperability architecture

Citation for published version (APA):

DOI:
10.1109/SAINT.2011.54

Document status and date:
Published: 01/01/2011

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher’s website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.

Download date: 06. May. 2020
Delay Performance in a Semantic Interoperability Architecture

Sachin Bhardwaj, Tanr Özçelebi, Richard Verhoeven and Johan Lukkien

Department of Mathematics and Computer Science,
Eindhoven University of Technology
P.O. Box 513, 5600 MB, Eindhoven, The Netherlands
{s.bhardwaj, t.ozcelebi, p.h.f.m.verhoeven, j.j.lukkien}@tue.nl

Abstract— A semantic interoperability architecture typically consists of collaborating heterogeneous networks and devices with different availability of resources such as bandwidth, processing power, memory, storage and energy. Due to utilization of ontology based information representation and communication formats, a gateway approach for achieving semantic interoperability becomes complex and may require long processing times. Especially when the constraints on acceptable response times in services with user interaction are considered, the end-to-end delay becomes an important performance criterion. In this paper, we make an analysis of the processing delay per device and the transmission delay per hop in a heterogeneous smart network. We identify the processes and the transmission links that dominate the end-to-end delay based on our experiments on a power-managed smart lighting scenario.

Keywords- end-to-end delay; heterogeneous networks; knowledge processor; smart space; smart lighting; semantic information broker; and semantic interoperability.

I. INTRODUCTION

The smart spaces concept is considered as a means to enhance the life standards of people and is being widely investigated nowadays. A smart space can be a heterogeneous network consisting of multiple networks of different device platforms. Low-capacity devices such as wireless sensors and actuators operate using very simple protocols and optimized message formats, allowing them to offer and consume only simple services. Devices with strict resource constraints satisfying the following criteria can be considered in this category: 1-25 MHz CPU clock frequency, 1-10 kbyte RAM, 1-128 kbyte programmable ROM, a communication bandwidth of several kilobytes per second typically using the frequency band of 433MHz-2.4 Ghz, a 2.0 – 2.5 Ah or less electric charge, and i/o devices such as sensors and actuators. In this work, a network that includes at least one such low-capacity device is called a low-capacity network, and a network that includes no low-capacity devices is called a high-capacity network. The role of a semantic interoperability architecture is to enable semantic information exchange between different device platforms and across heterogeneous networks, e.g. between a low- and a high-capacity network.

Recently, there have been efforts [1-4] to provide technology solutions for semantic interoperability, where the information describing the smart spaces and events therein can be interpreted automatically and accurately by all devices involved. Ontology based semantic information representation formats are commonly used in this area. A gateway node [4] that is capable of translating between the semantic information formats of different networks allows a low-capacity network to interoperate with a high-capacity network. However, performance becomes an issue using such a design. Firstly, the gateway node increases the number of hops in the end-to-end path between devices in different networks by one. Secondly, the processing at the gateway node is more complex than a generic purpose optimized (e.g. IP) gateway that runs on a single network layer protocol, introducing long processing times. Therefore, the end-to-end delays between collaborating devices in different networks become a key performance indicator for the overall system.

There are many applications that can make use of a gateway interoperability solution. For example, consider smart lighting [4], where the goal is to achieve and maintain a certain range of illumination in a surface area regardless of any interference from external light sources such as daylight. This is done through monitoring illumination in the activity surface area using light sensors and dimming the luminaries accordingly. In this case, excessive communication delay in the feedback channel from light sensors to light actuators can make the system unstable when combined with automatic and frequent light level adjustments. For instance, if the light sensor reports that the illumination in the surface area is below the desired range, the light actuator starts increasing its light output until it receives another feedback from the sensor, indicating that the desired light levels are achieved. When the delay is high, the illumination in the surface area can go above the desired range, which in turn causes the actuator to dim the light back down below the desired range. As a result, the illumination may start oscillating around the desired range in an unstable manner. On the other hand, if the sensor-to-luminary delay is known to the system, such unstable behavior can be easily avoided by changing the frequency of light adjustments.

In this paper, the end-to-end delay performance of the semantic interoperability architecture described above is investigated. Transmission and processing delays are analyzed on individual links and devices to identify bottlenecks and recipes for better application performance. Our experiments carried out in a power-managed smart lighting system [7] show that the worst case end-to-end delay that must be taken into account for stable system behavior can be in the order of hundreds of milliseconds.
The paper is organized as follows: In Section II, the related work and background of the proposed system is described. In section III, the proposed semantic interoperability architecture and the delay performance are introduced. In Section IV, the experimental use case scenario and results are presented. In Section V, conclusions are drawn and future research directions are suggested.

II. BACKGROUND

The focus of semantic interoperability research in the literature has been mostly on the design of smart space architectures and semantic interoperability over heterogeneous networks and devices [1-4], rather than performance analysis. The heterogeneous network in [4] is formed out of a low-capacity network of OSAS (Open Source Architecture for Sensors) [5] nodes and a high-capacity network of Smart-M3 (multi-vendor, multi-device, multi-domain) [6] nodes. The semantic information interoperability is achieved over the gateway, which can translate between the Smart-M3 ontology and the OSAS message format. This enables mashing-up and integration of information across all applications spanning from the embedded domain to Internet. Further studies and implementations of semantic interoperability between low and high capacity nodes are given in [6-7]. In [7], communication across different network platforms is achieved through wireless and wired communication protocols such as TCP/IP and IEEE 802.15.4 [8].

A. Smart-M3

The Smart-M3 architecture is flexible and modular, facilitating the use of various transport technologies, application development environments and programming languages. It introduces three main smart space components, by which the semantic information can be stored, removed and browsed: i) Knowledge Processor (KP), ii) Semantic Information Broker (SIB), and iii) Smart Space Access Protocol (SSAP). A KP is an entity that produces or consumes information according to the ontology relevant to its defined functionality. There are three KP types distinguished by data flow direction: producer KP, consumer KP and producer-consumer KP. A producer KP gets raw data from the physical environment and produces semantic information (e.g. whether the preferred light luminance levels are achieved or not in an activity space) to Smart-M3. A consumer KP can make queries or subscriptions in Smart-M3 and process the results. A producer-consumer KP can do both tasks. The suitable KP class to be used depends on the application requirements. A SIB is an entity, in which high level semantic information is stored and maintained. This information can be used by other devices in the smart space. The Smart Space Access Protocol (SSAP) is a simple set of primitives to insert, remove and access data in the SIB, and it can be used on top of transport technologies such as TCP/IP or Network on Terminal Architecture (NoTA) [9]. A KP can use SSAP for several transactions to the SIB such as join, leave, insert, remove, update, query, subscribe, and unsubscribe. As SSAP of Smart-M3 is transport layer independent, the same SIB can serve KPs over several transport technologies at the same time, while OSAS nodes can communicate only over the IEEE 802.15.4 protocol. Note that, the gateway node of [4] communicates with OSAS nodes and the Smart-M3 SIB over IEEE 802.15.4 and over TCP/IP, respectively, acting as a producer-consumer KP. Unlike the low capacity sensors and actuator devices in the OSAS network, a KP in the Smart-M3 network can communicate directly to the SIB.

III. SEMANTIC INTEROPERABILITY ARCHITECTURE

A semantic interoperability architecture enables seamless connection between nodes of a heterogeneous network, which consists of multiple networks utilizing different architectures and protocols. The semantic interoperability system architecture introduced by [7] is as shown in Fig. 1, where a number of producer KPs, consumer KPs join a SIB to form a smart space.

Fig. 1. Semantic interoperability architecture

A KP can produce information for the SIB or consume information available at the SIB using SSAP. A consumer-producer KP called the gateway KP (GWKP) acts as a bridge between low-capacity nodes such as sensor nodes (SN) and actuator nodes (AN) on the one hand, and the high-capacity smart space SIB on the other hand. The information gathered at the SIB from Smart-M3 KPs is stored using an ontology model [10], including the information gathered from the OSAS network through the GWKP. The information representation is implemented using Resource Description Framework (RDF) [11] triples. The RDF triples are represented by Subject (S), Predicate (P), Object (O) i.e. \{S, P, O\}.

A. Delay Performance

In evaluating the delay performance, we consider action (A) and reaction (R) times in the semantic interoperability architecture. Applications can be defined for the smart space such that an action of an entity (a device or a service) finds response with a certain reaction of another entity as shown in Fig. 2. The end-to-end delay from ‘entity X’ (action A) to ‘entity Y’ (reaction R) is a determining factor for the
minimum time distance between successive reactions. Based on our analysis in Section I, the time interval between any two consecutive reactions of the same type (e.g., light output adjustment) at ‘entity Y’ must be greater than the delay between A and R for stable operation.

In our semantic interoperability architecture, there are two different modes of message transmission: (i) direct messaging between a KP and a SIB, and (ii) message relaying between SN/AN devices and the SIB through the GWKP. The system is evaluated considering these two modes of transmission based on the end-to-end delay performance from A to R. In particular, we show how the system behaves for a given lighting scenario and occurrence of successive actions and reactions.

B. End-to-End Delay

The end-to-end delay is defined as the time taken for information to be processed and transmitted across a network from source to the destination. In the Smart-M3 architecture, the transmission delay from a KP to a SIB is given by \( \Delta t_{KPoSIB} \), and is calculated by (1), where \( t_{KPoSIB,Tx} \) and \( t_{KPoSIB,Rx} \) denote the measured message transmission and reception time instances. The delay for the GWKP to update and retrieve information at the SIB is given by \( \Delta t_{GWKPoSIB} \) and \( \Delta t_{SIBtoGWKP} \) respectively. The calculations are given by (2) and (3), where \( t_{GWKPoSIB,Tx} \) and \( t_{SIBtoGWKP,Tx} \), and \( t_{GWKPoSIB,Rx} \) and \( t_{SIBtoGWKP,Rx} \) denote the message sending and receiving time instances, respectively.

\[
\begin{align*}
\Delta t_{KPoSIB} & = t_{KPoSIB,Rx} - t_{KPoSIB,Tx} \\
\Delta t_{GWKPoSIB} & = t_{GWKPoSIB,Rx} - t_{GWKPoSIB,Tx} \\
\Delta t_{SIBtoGWKP} & = t_{SIBtoGWKP,Rx} - t_{SIBtoGWKP,Tx}
\end{align*}
\]

Similarly in OSAS architecture, the SN updates light intensity values to the GWKP with delay of \( \Delta t_s \), given by (4). Here, the measured message sending and receiving time instances are denoted by \( t_{s,Tx} \) and \( t_{s,Rx} \). The AN is updated by the GWKP with a delay of \( \Delta t_a \) as calculated in (5), where the measured message sending and receiving time instances are given by \( t_{a,Tx} \) and \( t_{a,Rx} \).

\[
\begin{align*}
\Delta t_s & = t_{s,Rx} - t_{s,Tx} \\
\Delta t_a & = t_{a,Rx} - t_{a,Tx}
\end{align*}
\]

The end-to-end delay (\( \Delta t_{e,a} \)) of the smart space is calculated based on a complete state of A to R. For example, an action taken by entity (KP) in Smart-M3 creates a reaction by AN in OSAS. This means that the update on AN in OSAS depends on information from KP in Smart-M3, which is a complete state required for the end-to-end delay (i.e. \( t_{e,a} \)) and given by (6). Similarly, if the reaction of AN is dependent on the action of SN then the end-to-end delay (i.e. \( \Delta t_{e,a} \)) is given by (7).

\[
\begin{align*}
\Delta t_{e,a} & = \Delta t_{KPoSIB} + \Delta t_{SIBtoGWKP} + \Delta t_a \\
\Delta t_{e,a} & = \Delta t_s + \Delta t_a
\end{align*}
\]

The calculation of end-to-end delay in (6) and (7) helps to estimate the time for the next successive action occurrence for KP and SN, respectively.

IV. EXPERIMENTAL SCENARIO AND RESULTS

We consider a scenario of power managed smart lighting [7] for experiments. The scenario is divided into two rooms with different priorities, called low priority room and high priority room, which are for low and high priority user activities, respectively. While the OSAS architecture is employed in the low priority room, the Smart-M3 architecture is used in the high priority room. In this scenario, the aim is to manage the overall power consumption due to lighting in these two rooms, i.e., keep it below a certain assigned power quota (12 Watts).

The Smart-M3 KPs are used for the high priority activities and report information of power consumption by the luminaries directly to the SIB. On other hand, the GWKP is used for the low priority activities and it subscribes to the SIB for the information on power consumption in the high priority room. The GWKP is also connected to the SNs and ANs. Therefore, it gets informed about the changes in the low priority room environment and can change its behavior. Semantic interoperability is achieved between the Smart-M3 and OSAS architectures by exchanging information between the aforementioned devices in the high and low priority rooms. The use case scenario is executed according to the following steps:

1. The GWKP and the other KPs in the low and high priority rooms join the smart space by joining the SIB.
2. The KP luminaries in the high priority room change their brightness to achieve the desired illumination levels in the high priority room. These luminaries can measure their power consumption and send this information to the SIB. Note that the power consumption depends on the contribution of external lighting to the room illumination.
3. The GWKP makes two subscriptions: one Smart-M3 subscription to the SIB and one OSAS subscription to the SN. The first subscription provides the GWKP with the power consumption information of luminaries in the high priority room, whereas the second subscription provides it with the illumination levels measured in the low priority room.
4. Based on the power quota and the power consumption at the high priority room, the GWKP calculates the remaining power budget for illuminating the low priority room.
5. The GWKP sends OSAS messages (light actuation commands) to the ANs in the low priority room. The specific AN hardware that we use [7] can update the luminary attached to it every 33ms.

6. The ANs start dimming their light outputs without a priori knowledge of how much increase or decrease in the light output is required. Due to lack of such knowledge and for achieving smooth light transition effects, the luminaries need to adjust light with small step sizes. Thus, in practice, multiple reactions (light adjustments) are needed per action. An AN can conclude that the desired light levels are achieved only after it receives a message from the GWKP that indicates this. If the reactions are too frequent (many light adjustments without knowing the actual status) or if the light adjustment step size is too large (illumination jumps from too low to too high and vice versa), this may result in instability. To avoid this, we must take into account the worst case behavior of end-to-end delay after an action, which is the contribution of this work. When the delay is large, the quota might be exceeded for a long time.

This is how the overall power consumption is controlled and kept below the power consumption quota at all times. The physical view of the system is shown in Fig. 3. The GWKP is connected with two SNs (SNa and SNb) and two ANs (ANA and ANb) over IEEE 802.15.4 and USB serial communication links, respectively. ANA and ANb are connected to the luminaries L1 and L2 for regulating the light output and for providing the desired level of illumination in the low priority room. In the high priority room, there are two KPs (KP1 and KP2) with luminaries (L3 and L4), which are capable of measuring the power consumption of the luminaries. The delay performance is evaluated based on the time between action-reaction pairs Aa-Ra and Ab-Rb of the scenario.

We consider two example use cases and measure end-to-end action-reaction delays. In the first use case, the external lighting condition changes in the high priority room (e.g., opening the blinds lets daylight in), allowing KP1 and KP2 to dim down the luminaries L3 and L4, reducing their power consumption. This situation increases the leftover budget for the low priority room, which means that AN1 and AN2 can now increase the brightness of L1 and L2 to desired levels. Therefore, actions of the KPs in the high priority room (Aa) can result in reactions by the luminaries in the low priority room (Ra).

Case 1:

Aa: External light changes in the high priority room. KP1 and KP2 update the power consumption information of L3 and L4 at the SIB accordingly.

Ra: The luminaries L1 and L2 are adjusted by AN1 and AN2 upon commands from the GWKP.

Aa → Ra: \( \Delta t_{\text{KPtoSIB}} + \Delta t_{\text{SIBtoGWKP}} + \Delta t_a \)

In the second use case, the illumination changes in the low priority room again due to changing external light (Ab), leading to subscription notifications from SNa and SNb to the GWKP and light commands from the GWKP to ANa and ANb. L1 and L2 are dimmed accordingly (Rb), satisfying the desired illumination levels. Note that the first use case covers both OSAS and Smart-M3 networks, while the second use case covers only the OSAS network.

Case 2:

Ab: External light changes in the low priority room. (SNa and SNb create an event to inform the GWKP.)

Rb: Luminaries L1 and L2 will react based on the commands from ANa and ANb for adjusting the light level.

Ab → Rb: \( \Delta t_b + \Delta t_a \)

The delay measurements are performed by capturing the network traffic that flows through the SIB while using a wired UTP (Unshielded Twisted Pair) network. The delay between a SN and the GWKP are measured by attaching both to an oscilloscope. The results of the measurements are shown in Table 1, while the cumulative probability distributions for the different delays are provided in Fig. 4. The measurements show that the delay between a KP and the SIB is rather large with a considerable variance. The communication from the KP to the SIB consists of an UPDATE request and the related CONFIRM response. The average delay of 176.71 milliseconds provides a major contribution to the total end-to-end delay. The
communication from the SIB to the GWKP consists of an INDICATION event from the SIB due to a subscription and results in a QUERY request from the GWKP, terminated by a CONFIRM response by the SIB. There is some variance in the communication delay and the average delay of 25.87 milliseconds might become problematic when the SIB has to inform and handle multiple subscribers. These two delays contribute to the end-to-end delay from a KP to the GWKP and the average sum is 202.58 milliseconds, including TCP/IP transmission and processing delay components. The communication delay components of \( \Delta_{GWKPtoSIB} \) and \( \Delta_{SIBtoGWKP} \) are equal on average, since both connections utilize TCP/IP connections on the same link.

![Cumulative Probability Distribution](image)

**Fig. 4.** Cumulative probability distribution of delays between different entities in the network.

The delay measurements for the communication between the SN to the GWKP shows a small variance, which is partially caused by variations in accessing the radio channel for transmissions. For the measurements, the SN was at a one hop distance from the base station and no duty cycle was used for the radio channel. For a more extensive wireless network configuration with duty-cycled radios, the delay characteristics would be more complex. The delay between the GWKP and the AN is currently not measured but estimated. The hardware platform to control the luminaries and the GWKP and the AN is currently not measured but estimated. The delay between network configuration with duty-cycled radios, the delay one hop distance from the base station and no duty cycle was for transmissions. For the measurements, the SN was at a partially caused by variations in accessing the radio channel the SN to the GWKP shows a small variance, which is

<table>
<thead>
<tr>
<th></th>
<th>( \Delta_{GWKPtoSIB} )</th>
<th>( \Delta_{SIBtoGWKP} )</th>
<th>( \Delta_a )</th>
<th>( \Delta_e )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>25.87</td>
<td>176.71</td>
<td>15.98</td>
<td>16.67</td>
</tr>
<tr>
<td>Std.Dev.</td>
<td>6.23</td>
<td>17.79</td>
<td>1.19</td>
<td>9.96</td>
</tr>
<tr>
<td>Minimum</td>
<td>12.30</td>
<td>145.55</td>
<td>13.28</td>
<td>0.00</td>
</tr>
<tr>
<td>Maximum</td>
<td>48.90</td>
<td>244.19</td>
<td>18.72</td>
<td>33.33</td>
</tr>
</tbody>
</table>

**TABLE I. DELAY MEASUREMENTS**

As the frequency of actions and reactions in a smart space is bound with limits measured by the end-to-end delay, we investigated the delay performance of a semantic interoperability platform. The experiments were carried out in a power managed smart lighting system test bed. Our analysis proved that the end-to-end delay can be significant enough to jeopardize the stability of collaborative services provided by the smart space. Furthermore, we were able to come up with a statistical upper bound on the worst case delay performance for the given experimental setup. One important conclusion is that KP-to-SIB and KP-to-GWKP transmission delays can be dominating in the end-to-end delay. Hence, they need to be optimized.

The present scenario is implemented on the smart lighting concept and end-to-end delay between smart space actions and reactions is used as the sole performance parameter. The experimental results presented show the importance of delay performance in smart spaces. As future work, this study will be extended to a performance analysis in terms of network traffic and scalability. Currently, we consider a smart space that serves only one application, whereas multiple applications will bring further challenges in terms of performance.

**ACKNOWLEDGMENT**

This work is supported by Smart Objects For Intelligent Applications (SOFIA) project and funded through European JTI ARTEMIS under the programme “Smart Environments and Scalable Digital Services”.

**REFERENCES**


