Semantic Coordination of Ambient Intelligent Medical Devices – A Case Study

Safdar Ali
Dept. of Intelligent Health Systems
Fraunhofer Institute for Biomedical Engineering
Ensheimerstr. 48, 66386, St. Ingbert, Germany
safdar.ali@ibmt.fraunhofer.de

Stephan Kiefer
Dept. of Intelligent Health Systems
Fraunhofer Institute for Biomedical Engineering
Ensheimerstr. 48, 66386, St. Ingbert, Germany
stephan.kiefer@ibmt.fraunhofer.de

Abstract— **In this paper, we describe a clinical laboratory scenario, where the next generation *Ambient Intelligent* (AmI) medical devices *semantically* coordinate with each other not only for the diagnosis of Pheochromocytoma and/or Neuroblastoma tumors, but also the forwarding of a higher level of interpreted results to a remote health information system, using a 3G mobile device as a gateway, to assist a health professional for meticulous diagnosis. These AmI medical devices are enriched with our SOA based middleware infrastructure, named *Semantic Medical Devices Space*, which supports *ontology* based semantic discovery of desired medical devices, and provides Semantic Web Service based interface for exchanging the measurement results.**

Keywords-semantic interoperability; semantic web services; ambient intelligence; medical devices

I. Introduction & Background

The pivotal role of Semantic Web and Web Services has been recognized for leveraging the legacy patient's healthcare information to machine interpretable, processable, and offerable as *services*, i.e. in ARTEMIS project [1]. Similar efforts, i.e. in SAPHIRE project [2] have been made to integrate the wireless medical sensor data with Hospital Information System (HIS) using a semantic interoperability platform for intelligent healthcare monitoring. The key problem with state-of-the-art medical devices is the *use* of vendor specific (proprietary) protocols to communicate with HISs and/or Laboratory Information Systems (LISs), which obstructs their communication with other medical devices, even though provided by the same vendor, when required in some scenarios. On the other hand, the SDOs (Standards Developing Organizations), i.e. IEEE, NIST, ISO, ASTM, have made various efforts for standardizing the medical device communication protocols, i.e. ISO/IEEE 11073 [3], but the medical device industry is still hesitant in adopting these communication standards, mainly for business reasons.

We have developed a SOA based middleware infrastructure using Semantic Web and Semantic Web Service (SWS) technologies, named *Semantic Medical Devices Space* (SMDS) [4], which enables the next generation *Ambient Intelligent* (AmI) medical devices to semantically *discover* the desired medical device(s), based on their *physical* and/or *functional* characteristics and use their service(s) for exchanging the measurement results and providing a higher level of interpretation of these results to realize the advanced health care scenarios, few of them are presented in [5][6][7][8]. SMDS is enriched with our own developed semantic discovery protocol and an integrated micro OWL Description Logic (DL) reasoning system to support *semantic discovery*, *knowledge querying & consistency checking* and *inference capabilities* on each AmI medical device. We define the AmI medical devices as the next generation medical devices which have:

- the ability to semantically *discover* and *match* the desired medical devices in a pure (un)structured Peer-to-Peer (P2P) network, based on their physical and/or functional characteristics. The *physical characteristics* (i.e. device/sensor vendor, device/sensor group/type etc.) are exhibited using a device ontology, while the *functional characteristics* (i.e. semantic web service methods, description etc.) are exhibited through service and domain/application ontologies.
- the ability to *offer* and *use* the semantic web services in the environment to help in achieving the plug & play interoperability.
- an *integrated* knowledge base, reasoning and querying system, which makes them *autonomous*.

Pheochromocytoma [9] is a rare tumor of the adrenal gland that causes too much release of hormones (epinephrine and norepinephrine) that regulate heart rate and blood pressure, while Neuroblastoma [10] is a common extracranial solid tumor in infancy, which develops from the nerve tissues. Both of these tumors, besides analyzing the blood samples, can be diagnosed by analyzing the levels of urinary catecholamines. In our laboratory scenario, a blood analyzer processes a particular patient's blood sample and waits to retrieve the urinary catecholamines of that patient from the urine analyzer, and then processes both of these results together to provide a higher level of interpretation about the disease status to the health professional.

In this paper, we have used the term semantic *coordination* instead of semantic *interoperability*, because the latter involves the *composition* and *planning* phases of semantic web services, which are not required/implemented yet for the resource constrained medical devices in our scenarios. The rest of the paper is organized as follows: Section 2 gives a brief description of semantic discovery process and how the

This work is partially supported by European Commission within the context of SmartHEALTH IP Project, contract 016817, priority FP6-2004-IST-NMP-2; http://www.smarthealthip.com

Digital Object Identifier: 10.4108/ICST.PERVASIVEHEALTH2009.5975
http://dx.doi.org/10.4108/ICST.PERVASIVEHEALTH2009.5975

reasoning procedure is done. Section 3 illustrates the laboratory scenario, while Section 4 gives the implementation details. Section 5 gives the performance evaluation results, while Section 6 outlines conclusions and future work.

II. Semantic Discovery of Ambient Intelligent Medical Devices

In order to support semantic discovery of medical devices, we have developed a lightweight HTTP based *Semantic Medical Device Discovery Protocol* (SMDDP), which enriches the medical devices to send/receive *semantic queries* over an unstructured Peer-to-Peer (P2P) topology based network of AmI medical devices. Fig. 1 shows the overall workflow of SMDDP protocol, where every medical device consists of two processes (like agents), namely smddp client and smddp server to support querying and providing information to the requesting medical device(s). When an AmI medical device needs to search for the desired medical device(s),

- the *smddp client* broadcasts the query (with concrete conditions) using a multicast UDP protocol throughout the network.
- the *smddp server* on each medical device receives the query, processes it with its knowledge base, and if matched, returns the results back to the requesting medical device(s).

The requesting client medical device attaches a *callback HTTP URI* to the request, so that every smddp server could send the response back to that endpoint using traditional HTTP protocol, which is augmented with security mechanisms (basic authentication, digest or HTTPS).

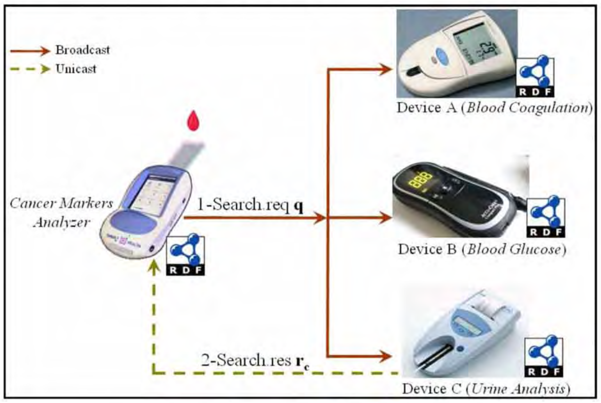
Broadcast
Unicast
Cancer Markers Analyzer
RDF
1-Search.req **q**
2-Search.res $\mathbf{r}_c$
Device A (*Blood Coagulation*)
RDF
Device B (*Blood Glucose*)
RDF
Device C (*Urine Analysis*)
RDF

Figure. 1 Overall workflow of SMDDP protocol

A. *SCENT: Semantic Medical Device Language for N-Triples*

In order to express the semantic query, we need a query language, i.e. SPARQL [11] and further to process it, we require a query engine, i.e. SPARQL query engine [12], which is clearly not possible to run over a resource-constrained medical device. To cope with this problem, we have developed a simple alternative by modifying the EBNF (*Extended Backus-Naur Form*) of N-Triples [13], which is a line-based, plain text format and simple grammar to encode the basic standardized RDF notation. We call this extended version of N-Triple patterns language as SCENT (*Semantic Medical Device Language for N-Triples*), which represents a subset of SPARQL expressiveness and its simplicity makes is possible to be processed by resource-constrained medical devices. Contrary to SPARQL, SCENT doesn't support *PREFIX* for the predicates and does not allow keywords, i.e. *FILTER*, *OPTIONAL* and other additional options. Table 1 shows a comparison between original N-Triples syntax and the SCENT language syntax specifications, where we modified the original *subject* and *object* productions with the inclusion of *variable* (starting with ? sign) which now allows to use variables together with absolute URIs at the places of *subject* and/or *object* of a triple.

TABLE 1: COMPARISON OF ORIGINAL N-TRIPLES AND SCENT SYNTAXES

Original N-Triples Specification	subject ::= uriref | nodeID object ::= uriref | nodeID | literal
Modified N-Triples Specification (SCENT)	subject ::= uriref | nodeID | **variable** object ::= uriref | nodeID | literal | **variable** **variable ::= '?' name**

LISTING 1: AN EXAMPLE OF SMDDP REQUEST MESSAGE

```
SEARCH ?d smddp/1.0
SeqN: 20
Content-Type: application/de.fhg.ibmt.scent
Content-Length: 400
Callback-Uri: http://192.168.2.11/smddpcallback

?d <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
   <http://www.ibmt.fhg.de/onto/2008/04/md#MedicalDevice> .
?d <http://www.ibmt.fhg.de/onto/2008/04/md#hasGroup> ?g .
?g <http://www.ibmt.fhg.de/onto/2008/04/md#deviceType> "UrineAnalyzer" .
?d <http://www.ibmt.fhg.de/onto/2008/04/smds#hasMeasurementOf> ?p .
?p <http://www.ibmt.fhg.de/onto/2008/04/smds#patHisId> "23KD2008" .
```

Listing 1 shows the complete SMDDP request message that is disseminated over the network of medical devices, as shown in Fig. 1, containing a query comprising of SCENT conditions (patterns). The query (**q**) is described using *MeDO*, a small medical device ontology for describing the physical characteristics, as shown in Fig. 2, while the functional characteristics are described using *SmdsOnto* ontology, as shown in Fig. 3, in conjunction with W3C SAWSDL [14] standard to semantically annotate the service description (WSDL) files of each AmI medical device .

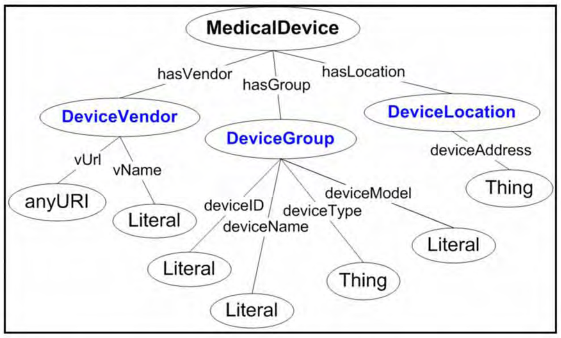

Figure. 2 MeDo ontology for describing physical characteristics

Digital Object Identifier: 10.4108/ICST.PERVASIVEHEALTH2009.5975
http://dx.doi.org/10.4108/ICST.PERVASIVEHEALTH2009.5975

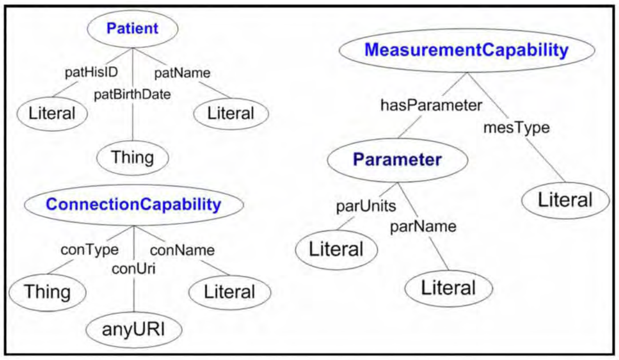

Figure. 3 SmdsOnto ontology for describing functional characteristics

In Listing 1, the Line 1 shows the **SEARCH** command and the variable name (**?d**) together with the protocol version number. **SeqN** (Line 2) is a request sequence number used to detect duplicate request messages from client devices. **Content-Type** (Line 3) shows the MIME type of the message body. **Content-Length** (Line 4) shows the length of the message body in bytes, and **Callback-URI** (Line 5) shows the client's HTTP endpoint where the server must send back the response. The second part of the request message shows the SCENT conditions (matching patterns), describing that a client device is looking for a *medical device* whose *type* is *Urine Analyzer* having the *measurement* results of a patient with the *id* of *23KD2008*. The query engine on every AmI medical device processes the semantic query and if it *matches* with the provided conditions (a measurement record is found), it replies back to the client on the given HTTP callback-URI. As shown in Fig. 1, only the matched urine analyzer responds back to the semantic query broadcasted by cancer marker analyzer (CMA).

LISTING 2: AN EXAMPLE OF SMDDP RESPONSE MESSAGE

```
POST /smddpcallback HTTP/1.0
Host: 192.168.2.11
SeqN: 20
Content-Type: application/de.fhg.ibmt.smiddel
Content-Length: 561

<?xml version="1.0" encoding="UTF-8"?>
<smiddel xmlns="http://www.ibmt.fhg.de/smds/2008/04/smiddel"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:schemaLocation="http://www.ibmt.fhg.de/smds/2008/04/smiddel
   http://www.ibmt.fhg.de/smds/2008/04/smiddel.xsd">

   <medevice uri="urn:uuid:urisys">
      <wsurl>http://192.168.2.25:8086/csoap/UrisysService</wsurl>
      <wsdl>http://192.168.2.25:8086/csoap/UrisysService/wsdl/urisys.wsdl</wsdl>
      <sawsdl>http://192.168.2.25:8086/csoap/UrisysService/wsdl/urisys.sawsdl</sawsdl>
   </medevice>
</smiddel>
```

Listing 2 shows the complete SMDDP response message sent back from a urine analyzer to the CMA device. The most important tags in this message are the URIs for the web service methods invocation, its WSDL and SAWSDL files. The reason to include three URIs is the lack of compilers (Java/C#) for the resource-constrained devices, and that's why we had to pre-create the client stubs for medical devices to invoke the web service methods available at the URI given in **<wsurl>** tag. However, the URI given in **<wsdl>** tag is for the information systems having enough computing power and compilers to process the WSDL *on-the-fly* and could dynamically create the web service client stubs. Similarly, the URI given in **<sawsdl>** tag facilitates the medical devices and information systems to dynamically process the *sawsdl* file and *find* the desired method to invoke for a particular context-aware application.

B. *Semantic Query Processing & Reasoning*

Fig. 4 shows the overall architecture of a small but powerful integrated semantic query processing & reasoning system [15] that we have developed for the resource-constrained AmI medical devices. It includes:

- The *Query Processor*, which is responsible for the tasks of *extracting* the SCENT conditions/patterns from the query, knowledge base (KB) *loading* and then finally resolving the query by *matching* the KB triples against these conditions.
- μOR, a micro OWL DL Reasoner that is capable of making inferences based on the provided KB and the loaded domain ontologies, e.g. MeDO and SmdsOnto.

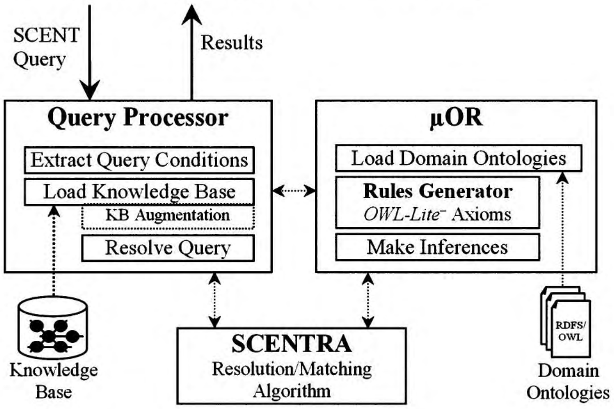

Figure. 4 Semantic query processing system & reasoning engine

- The SCENTRA algorithm is a simple *variables' unification* algorithm for the SCENT conditions/patterns matching & resolution, and commonly used by the Query Processor and μOR. The dominating worst-case time complexity of SCENTRA algorithm in the whole reasoning process is ***O(m.n.p)***, where *m* is the total number of conditions, *n* is the total number of KB triples, and *p* is the total number of variables used in SCENT conditions.

III. LABORATORY SCENARIO

In this section, we describe the healthcare scenario of Pheochromocytoma and/or Neuroblastoma screening/diagnosis using our SMDS middleware infrastructure, running on every AmI medical device. Fig. 5 shows the complete interaction of the AmI medical devices and the forwarding of the final results to the remote SmartHEALTH Information System (SIS). The SmartHEALTH Cancer Markers Analyzer (CMA) processes the blood sample of a patient and for the completion of the screening/diagnosis, it requires the urinary catecholamines levels of that patient. In order to do this, the CMA searches the network by broadcasting a query **q**, as shown in Listing 1, which specifies the discrete conditions for the desired urine analyzer. Although, there are two other medical devices in the

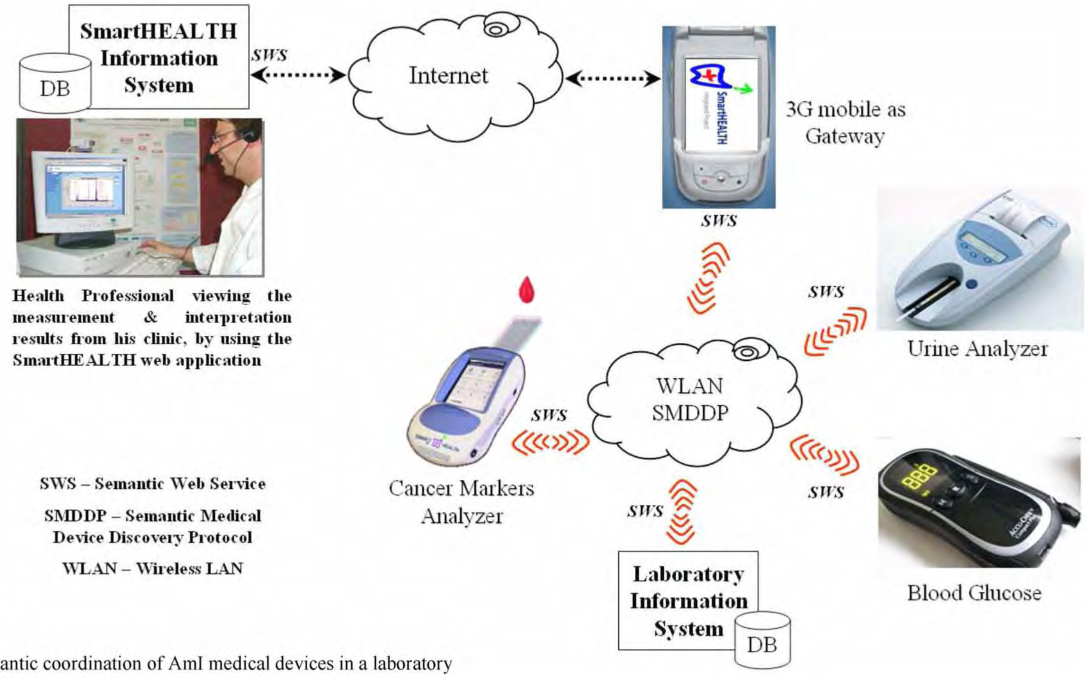
SmartHEALTH Information System
DB
SWS
Internet
3G mobile as Gateway
SWS
Health Professional viewing the measurement & interpretation results from his clinic, by using the SmartHEALTH web application
SWS
Urine Analyzer
WLAN SMDDP
SWS
SWS
Cancer Markers Analyzer
SWS
SWS – Semantic Web Service
SMDDP – Semantic Medical Device Discovery Protocol
WLAN – Wireless LAN
Blood Glucose
Laboratory Information System
DB

Figure. 5 Semantic coordination of AmI medical devices in a laboratory

environment, but only urine analyzer responds to the query of CMA, as it has performed the urine analysis of patient with the id *23KD2008*.

In the second step, the CMA calls the web service method of the urine analyzer to retrieve the urine analysis results of that patient and then computes the overall status of the tumor by comparing both blood & urine analyses. After computing the results and making a higher level of interpretation, the CMA searches the local network for a gateway device, which offers a semantic web service to forward the final results to the SIS. The CMA broadcasts a query about a device having a 3G internet connection with the SIS, in the similar way as it was broadcasted at first stage for the urine analyzer. As a result, the matched mobile gateway device responds to the CMA with its URIs, and the CMA downloads the *sawsdl* file from gateway device, extracts the method name using SAWSDL API, calls that method by sending the measurement and interpreted results. The mobile gateway device further calls the web service method of SIS to send the measurement and interpreted results. After the interpreted results are received on SIS side, the responsible health professional is informed in case of emergency (through an automatic alert, i.e. SMS/fax) about these results, so that he could login to the web portal of SIS site from his clinic (or even from home) and could view the measurement results along with a higher level of interpretation.

A. *Security Concerns*

This whole procedure of exchanging the measurement results between AmI medical devices, forwarding the sensitive information from CMA to mobile gateway device and then further to SIS is fully secured with an *end-to-end* security solution, developed using WS-Security [16] specification. We are using Verisign's device certificates in order to enable the AmI medical devices to *encrypt/decrypt* and/or *sign/verify* the exchanged messages among each other with the appropriate public and private keys. Different security levels have been defined, which can be configured according to the requirements of a particular healthcare application. After receiving the measurement results, the SIS server automatically recognizes which AmI medical device has sent these results, in order to keep track/log of each AmI medical device.

IV. IMPLEMENTATION PLATFORM

We have developed the complete SMDS framework and its components using Java™ programming language to realize the pervasive healthcare scenarios. For filtering and parsing of RDF information, we used Megginson's RDF Filter [17] and Piccolo [18] XML parser respectively. The beauty of Megginson's RDF Filter is it's small-size (12Kb) and support of automatic loading of ontologies mentioned with <owl:imports> statement. Piccolo is one of the fastest *SAX2* compliant XML parsers and has a small memory footprint.

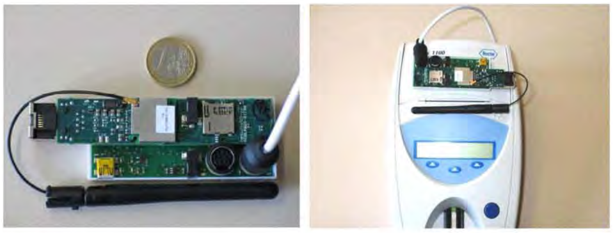

Figure. 6 a) Gumstix Platform b) Urine Analyzer from Roche

Digital Object Identifier: 10.4108/ICST.PERVASIVEHEALTH2009.5975
http://dx.doi.org/10.4108/ICST.PERVASIVEHEALTH2009.5975

As a hardware platform to test our querying and reasoning system, we selected Gumstix [19] XL6P motherboard with wireless network card and Embedded Linux operating system, as shown in Fig. 6a. A device driver for every medical device was developed in order to enable *serial* communication of medical device with its associated Gumstix, one of them is shown in Fig. 6b with Urine Analyzer (UA) 1100 device from Roche Diagnostics®. These Gumstixs enrich the medical devices, involved in the Laboratory scenario, with the capabilities of processing of semantic queries and reasoning, which are integral parts of *semantic discovery* and *semantic coordination* of AmI medical devices.

V. Performance Evaluation

We have tested the performance of our semantic query processing and reasoning system in comparison with a couple of other small reasoning systems, namely Pocket KRHyper [20] and Bossam [21]. We used a Windows XP system with Pentium® 4, Intel® 2.40 GHz processor, 1.5Gb RAM to perform 10 different tests of loading different domain ontologies with varying sized knowledge bases. Fig. 7 shows that the runtime memory size of complete µOR is just **24Kb**, which is significantly less than the memory sizes of Bossam (**750Kb**) and Pocket KRHyper (**245Kb**).

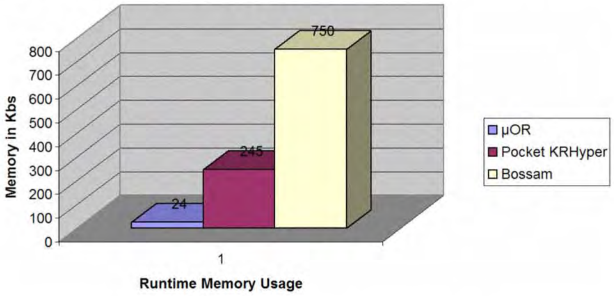

Figure. 7 Comparison of runtime memory usage

Similarly, Fig. 8, Fig. 9, and Fig. 10 show the comparison of times taken for ontology loading/conversion, knowledge base loading and overall reasoning in different reasoning systems respectively. Because the Gumstix has Marvell® PXA270 processor with 600MHz speed, which is a ratio of 4:1 to 2.40 GHz processor, so all the times calculated for µOR on PC will be of 1:4 ratio to Gumstix platform.

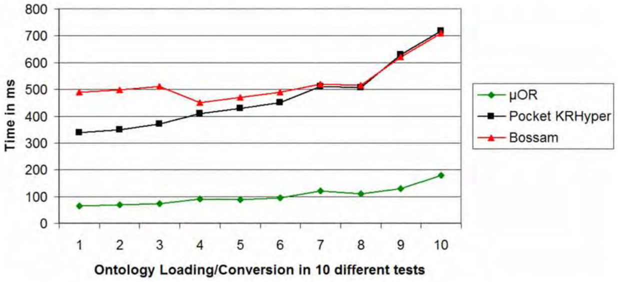

Figure. 8 Comparison of time taken for ontology loading/triples conversion

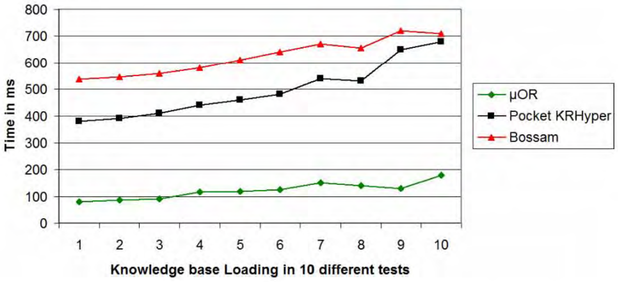

Figure. 9 Comparison of time taken for knowledge base loading

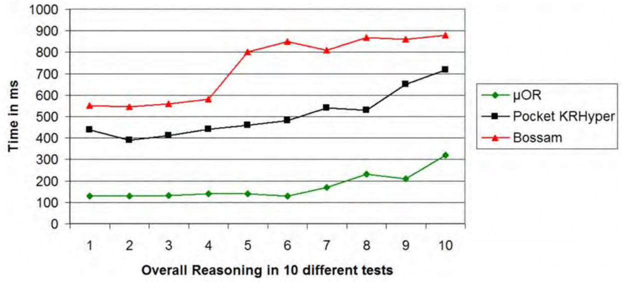

Figure. 10 Comparison of time taken for overall reasoning

VI. Conclusions & Future Work

In this paper, we have presented an advanced scenario of semantic coordination of AmI medical devices in a laboratory environment. Our SOA based middleware infrastructure, SMDS provides a platform for developing the next generation AmI medical devices, which are fully *autonomous* and enriched with integrated knowledge base querying and reasoning system. Our approach significantly affects the behavior of medical devices in an ambient intelligent environment, and leads to devise even advanced healthcare scenarios. In addition, our infrastructure provides semantic web services as a communication mechanism among AmI medical devices, which inherently resolve the platform heterogeneity problem, allowing diversified medical devices to be included in the network. The above performance/memory results show that the low memory/computing requirements of µOR make it feasible to be used on any resource-constrained medical/mobile device. The current implementation of µOR supports *Connected Device Configuration* (CDC[1]) ver. >= 1.0 compliant devices only, because of using `HashSet` and `HashMap` classes of Java for better performance, which are missing from the specification of CLDC[2] and MIDP[3] compliant devices. However, we will provide a substitutional version in future which will support both CLDC and MIDP devices as well.

Currently, we have used our own developed service/domain ontologies to semantically annotate the web services of AmI medical devices to have a proof of the concept, but in future,

[1] http://java.sun.com/javame/technology/cdc

[2] http://java.sun.com/products/cldc/

[3] http://java.sun.com/products/midp/

Digital Object Identifier: 10.4108/ICST.PERVASIVEHEALTH2009.5975
http://dx.doi.org/10.4108/ICST.PERVASIVEHEALTH2009.5975

we will investigate using standard service ontologies of OWL-S[4] and/or WSMO[5] together with SAWSDL for wide industry compliance.

ACKNOWLEDGMENT

We would like to thank Prof. Dr. Norbert Graf (University Hospital of Saarland, Homburg, Germany) for discussing the laboratory scenario of Pheochromocytoma & Neuroblastoma screening.

REFERENCES

[1] Asuman Dogac et al., "Artemis: Deploying semantically enriched web services in the healthcare domain", Elsevier Journal of Information Systems, Volume 31, Issues 4-5, The semantic web and web services, June-July 2006.

[2] Asuman Dogac et al., "SAPHIRE: Intelligent healthcare monitoring based on semantic interoperability platform - pilot applications", Proceedings of IEEE Communications-Special Issue on Telemedicine and e-Health Communication Systems, June 2007.

[3] ISO/IEEE 11073, Point-of-care medical device communication standard, http://www.iso.org/iso/iso_catalogue/catalogue_tc/cat-alogue_detail.htm?csnumber=36347

[4] Safdar Ali, Stephan Kiefer, "Semantic medical devices space: an infrastructure for the interoperability of ambient intelligent medical devices", Proceedings of IEEE-ITAB conference in Greece, October 2006.

[5] Heikki Helin et al., "CASCOM - context-aware health-care service co-ordination in mobile computing environments", ERCIM News num. 60.

[6] Sascha Ossowski et al., "Agent-based semantic service discovery for healthcare: an organizational approach", IEEE intelligent Systems, vol. 21, no. 6, pp. 11-20, 2006.

[7] Tim Finin et al., "A pervasive computing system for the operating room of the future", Mobile Networks and Applications, V.12, Issue 2-3, March 2007.

[8] Safdar Ali, Stephan Kiefer, Uribarren Aitor, and J. Parra, "Applications of ambient intelligence in medical devices and clinical environments", IEEE international conference of E-Medical Systems, Morocco, 2007; ISBN: 9954-8905-0-5.

[9] Medline Plus Medical Encyclopedia, http://www.nlm.nih.gov/-medlineplus/ency/article/000340.htm

[10] Medline Plus Medical Encyclopedia, http://www.nlm.nih.gov/-medlineplus/ency/article/001408.htm

[11] SPARQL Query Language for RDF, http://www.w3.org/TR/rdf-sparql-query/

[12] SPARQL Query Engine; http://sparql.sourceforge.net/

[13] W3C Recommendation, RDF Test Cases, 10 February 2004, http://www.w3.org/TR/rdf-testcases/#ntriples

[14] W3C SAWSDL, A mechanism to semantically annotate the Web Service Description Language files, http://www.w3.org/-2002/ws/sawsdl/

[15] Safdar Ali, Stephan Kiefer, "µOR – A micro owl description logic reasoner for ambient intelligent devices", 4th International Conference on Grid & Pervasive Computing, 2009 (to appear).

[16] WS-Security Specs; http://www.oasis-open.org/committees/wss/

[17] RDF Filter, http://rdf-filter.sourceforge.net/

[18] Piccolo XML Parser for Java, http://piccolo.sourceforge.net/

[19] Gumstix Miniature Computers, http://www.gumstix.com

[20] T. Kleemann, A. Sinner, "KRHyper - In your pocket, system description", In proc. of conference on automated deduction, LNAI 3632, pp. 452-457, 2005(CADE '05)

[21] J. Minsu, J. Sohn, "Bossam: an extended rule engine for owl inferencing", Proceedings of RuleML 2004

[4] http://www.w3.org/Submission/2004/SUBM-OWL-S-20-041122/

[5] http://www.w3.org/Submission/WSMO/

Digital Object Identifier: 10.4108/ICST.PERVASIVEHEALTH2009.5975
http://dx.doi.org/10.4108/ICST.PERVASIVEHEALTH2009.5975