Distributed and Passive Web services discovery middleware for Pervasive services at the edges of Internet

Abdul Haseeb, Mihhail Matskin
KTH, Royal Institute of Technology, Sweden
Email: {ahaseeb, misha}@kth.se

Peep Küngas
University of Tartu, Ülikooli 18, 50090, Tartu, Estonia
Email: {peep.kungas}@ut.ee

***Abstract*—The advent of mobile computing devices and development of wireless and ad hoc networking technologies as Bluetooth, RFID etc has led to the growth of infrastructure-less pervasive environments. Most often, these environments lie at the edges of Internet, i.e., they are disconnected or sparsely connected to rest of the world. In order to exploit the access to such edges of Internet, an interoperability middleware, capable of dealing with lack of communication infrastructure, is needed. In this paper, we propose a solution that synergizes P2P technology, message queuing support and a passive distributed UDDI to exploit Web services in infrastructure-less edges of Internet. We abstract communication heterogeneity and prove that passive communication mode performs better than active mode of communication (i.e., existing solutions in literature) in terms of Web services discovery.**

I. INTRODUCTION

Edges of Internet represent disconnected or sparsely connected sub-networks in a cascaded flow/collection of networks and associated technologies. These edges can range from wired to Wi-Fi and to ad hoc networks (for instance: mobile networks, war front networks, fire fighting networks, futuristic shopping malls, robot swarms etc.). In other words, we view a huge number of heterogeneous devices pervasive in everyday life (ranging from high-end servers to mobile devices and resource-constrained entities). These pervasive devices create communication clusters in multiple interacting yet independent networks. In some cases (due to mobility) these environments remain in isolation and disconnected from rest of the world, thus forming edges of Internet.

There is a need to exploit the access to such diversified natured and disconnected network environments without any manual intervention, i.e., an interoperability platform to glue various edges of Internet together. Interoperability of edges of Internet must deal with connectivity aspects (internal and connectivity with Internet). Internal connectivity can be either based on direct message passing or via shared memory. Connectivity with Internet, however, depends upon the openness of the system either using available connectivity or by using disconnected reconciliation-mode using mediators.

Web services technology has become a standard for interoperability problems. It relies on the ability to locate and acquire specified services. For such purposes UDDI provides the functionality to search for Web services. Existing UDDI technology uses a centralized repository to store pointers to registered Web services. Centralized approaches have many well-known and discussed drawbacks [8]. Moreover it is not viable to exploit centralized/semi-centralized solutions in infrastructure-less environments. Current solutions [16, 17] of service discovery in infrastructure-less environment are primarily broadcast based and inefficient in network resource utilization. These solutions do not address the issues of scalability and network-wide reachability. Moreover, issues of mobility and adaptability of protocols to dynamic environments is not addressed.

Apart from a decentralized UDDI requirement, there is a need to establish asynchronous interactions between entities. By using techniques developed as part of traditional Message Oriented Middleware (MOM), asynchronous messaging can be built on top of synchronous interactions by introducing a queuing system for storing and forwarding of messages. We refer to such queuing system as message post boxes. Mediation is useful in scenarios where P2P communication between entities is not possible.

In this paper we propose and evaluate an interoperability middleware that synergizes P2P technology, message queuing support and distributed UDDI to exploit Web services in infrastructure-less edges of Internet. This paper gives a detailed evaluation of a passive distributed UDDI, a loosely coupled distributed UDDI for infrastructure-less communicating and heterogeneous systems. We use robot swarms [3] for our evaluation, but the proposed principles are applicable to any infrastructure-less environments.

The rest of the paper is organized as follows. In Section II, we analyze Web services solutions in pervasive environments; Section III elaborates a representative edge. In Section IV and Section V, we present and evaluate our proposed distributed Web services discovery solution.

II. WEB SERVICES AND PERVASIVE ENVIRONMENTS

Pervasive environments can be classified into infrastructure-based and infrastructure-less environments. We focus on Infrastructure-less/ad hoc environments. Examples of such environments can range from fire fighting, war-front activities, robot swarms to space exploration research.

Service discovery architectures like Jini [12], Salutation and Salutation-lite [13], UPnP [14] and Service Location Protocol [15] have been developed in the past to efficiently discover Web services in infrastructure-based environments. But these discovery architectures rely on a central lookup for service registration and discovery. Central lookup is inappropriate in ad hoc environments due to its dependence on a central point/node. Moreover the central point/node can be mobile and unreachable during certain points of system execution.

Copyright (c) IARIA, 2010 ISBN: 978-1-61208-105-2

Solutions [16, 17] for service discovery for infrastructure-less environments primarily utilize broadcast-driven nature of the underlying ad hoc network. However, it was shown by [18] these approaches are not efficient and scalable for discovery in large-scale environments. Thus researchers tried a leverage P2P technology for distributed service discovery [2]. DUDE (Distributed UDDI Deployment Engine) [7] was among the earlier efforts that used distributed hash tables as rendezvous mechanism between distributed service registries. However, their approach cannot cope with high dynamism, mobility and in scenarios where entities cannot communicate in a P2P fashion. For infrastructure-less environment of mobile users/agents a messenger approach [9] was proposed as well, which provided mechanism for dynamic management of distributed UDDI in the absence of communication infrastructure. Basis philosophy of messenger is to update cached Web service descriptions at UDDI when a mobile user/agent comes in communication coverage area of some UDDI. This approach suits best when entities have unpredictable coverage, however, the solution did not consider collaborative service discovery. Similarly a proxy node at the edge of network to serve Web services to mobile users was proposed in [19]. But it dependence on wired infrastructure makes it inappropriate for infrastructure-less environments.

In summary, infrastructure-less environments are attributed by lack of stable connectivity [3, 9], mobility, heterogeneous communication capabilities [3] and lack of P2P communication. This puts forward the requirement for a solution which can overcomes the lack of availability of both service requesters and service repositories at the same time, asynchronous and mediator-based communication [5] for scenarios where P2P communication is not possible.

Our proposed methodology synergizes P2P technology, message queuing support and distributed UDDI to exploit Web services in infrastructure-less edges of Internet. Our proposed solution assumes environment “mailboxes” or “message postboxes” (for instance RFID tags), which can be used for communication when P2P communication between entities is not possible. Our solution also uses a passive mode of communication and service discovery. To the best of our knowledge, no earlier solution has used a passive mode of communication. In our evaluation we prove the effectiveness of passive communication mode in terms of network load while maintaining an acceptable latency.

III. SWARM OF ROBOTS – A REPRESENTATIVE EDGE

We take swarm of robots as a representative use-case/edge [3]. The edge comprises of an environment of low-cost robots operating in a dynamic environment. There is no assumption on the availability of wireless capability of robots - i.e., few robots have wireless capability, while the others use RFID tags for reading/writing data. Our objective is to establish interoperability of swarm both with environment entities and with outside world in a symmetric way.

We use Web services for exposing robotic functionality. Thus each robot is abstracted to a Web service interface. Robotic actions are based on a local knowledge-base and collection of swarm knowledge is represented as a knowledge gateway. Knowledge gateway is assumed to be a server, equipped with wireless capability, which can provide resource-intensive services to swarm entities. Apart from that any swarm entity with connectivity to outside world can serve as swarm gateway. We do not assume wireless capability on each robot; however, it is a reasonable assumption to have at least one robot with wireless capability that can communicate to Internet (see Figure 1).

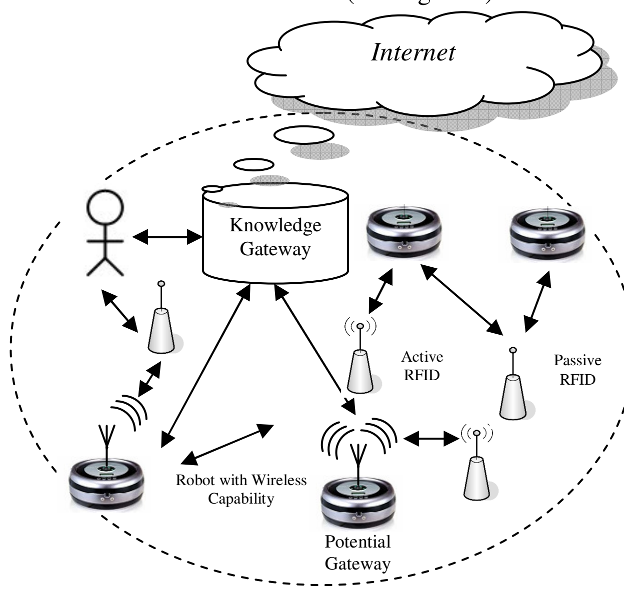

Figure 1. Swarm of robots – a representative edge.

IV. DISTRIBUTED SERVICES DISCOVERY MIDDLEWARE

We take the basic design philosophy from JXTA [1] - a P2P stack and XML-based set of peer platform protocols and services. In contrast to JXTA not only does our middleware rely on communication capabilities on entities, but also due to dynamism (caused by entity mobility) it supports dynamic joining of entities in different clusters during execution.

A. *Conceptual classification of entities*

Entities are classified in the following manner:

- **Rendezvous computing entities (RCE)** are the entities with Wi-Fi (or some other high-level) connectivity which enables them to communicate outside a domain or to external world. RCE serve for dissemination of requests from one cluster[1] of entities to another as routing peers [2]. RCE can create a virtual overlay network with other RCE (using Rendezvous Peer View protocol – RPV) to serve clusters [6] around mediators.
- **Edge computing entities (ECE)** are the entities those can’t communicate in a P2P fashion with other entities and require mediation for their communication.

[1] We refer to entities around MR as *entity-group/cluster*. I.e. Set of entities that can have a pair-wise communication among each other (P2P or via MR).

Copyright (c) IARIA, 2010 ISBN: 978-1-61208-105-2

- **Message Relays (MR)** are the mediators (e.g. RFID tags) that serve ECE to communicate their messages.

For the sake of clarity, we elaborate few essential components. Detailed discussion about the architecture can be read in our previous work [4, 20]:

- **Message Transport Layer (MTL)** is an implementation of asynchronous communication channels. Using MTL, entities and associated services can join and leave at any point of system execution, thus achieving loose coupling.
- **Local Service Registry (LSR)** is a local cache for Web service discovery. For LSR we implemented a light-weight UDDI [4]. Incoming request are first searched from LSR before being cached to **Query Response Cache (QRC)**. QRC caches incoming service requests and propagates back response messages using MTL.
- **Entity Discovery Registry (EDR)** serves as a record for discovered entities. EDR is used to create a semantic topology based on set of services an entity advertises and its expertise (i.e., semantics of services an entity offers).

B. Semantic Query Propagation

Notion of RCE, RPV protocol and dissemination of messages from one cluster to another provides quick dissemination of messages across clusters, while MR performs dissemination of messages within a cluster. To incorporate semantics and to perform semantic-based query propagation to appropriate entities, we use a model in which entities publish their *expertise* (semantics of services) in the network along with their exposed Web services.

Basic philosophy of creating a semantic topology is to re-route the query to an entity which is likely to answer the query instead of broadcasting or sending the query to random entities. For this purpose we use a shared common ontology. Entities publish their expertise in the network and the knowledge of an entity about the expertise of other entities forms a virtual semantic topology. According to our definition, an entity knows another entity in semantic topology if it can compute the semantic distance of a query to target entity's published expertise. Examples of expertise can be *Weather Information Services*, *Cleaning Services* etc.

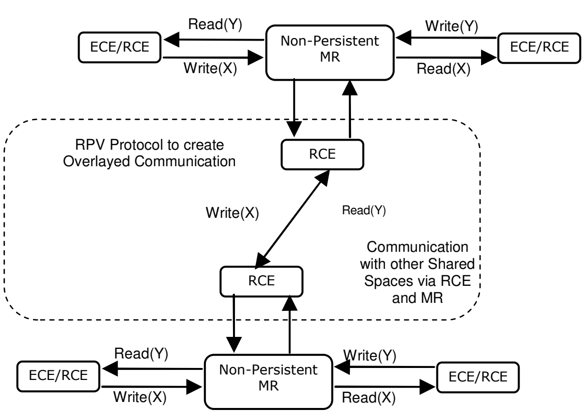

Figure 2. RCE overlay formation & communication between ECE & MR

C. Active and Passive communication mode

In term of communication, entities can follow two modes of communication.

1. Active Communication Mode

- Web service descriptions are pushed to other entities/MR.
- Active mode corresponds to a normal Web service publishing to UDDI.
- Service descriptions are first pushed inside a cluster via MR and later RCE disseminates them to rest of network.

2. Passive Communication Mode

- In passive communication mode a Web service discovery occurs when an entity's (RCE/ECE) request is answered by some entity/MR. In other words, entities don't publish their service descriptions unless requested.

MR are considered as non-persistent communication white-boards, which are prone to conflicts in case of concurrent access. In order to avoid conflicts entities employ a resource locking mechanism.

- **Conflicting mode** refers to a greedy mode in which entities don't wait for another entity to release MR.
- **Conflict-resolution mode** refers to a mode in which an active push or a passive service request locks the MR. Lock is released when either of the condition is true

1) Passing of service description or message request to another entity (i.e., at-least one entity has read the initiator's message).
2) Time-out

Due to constant mobility (and entities being out of communication range/coverage), lock to MR is released when at-least one entity has the initiator's message.

D. Entity and Web services registration

Upon a system bootstrap, entities (RCE and ECE) only have their own published Web services in LSR. Similarly their EDR is empty as well. Thus entities perform a registration process. For active communication mode registration includes Web services descriptions and a set of entity's expertise. However for passive communication mode registration only includes presence and endpoint related information.

Registration marks MR as an information container for services and entities surrounding it. Latter serves for discovery of entities when entities receive a list of other registered entities from the MR. This is a *naïve* epidemic way of discovering other entities in the environment.

RPV protocol enables RCE to organize them into a virtual overlay as shown by Overlayed communication in Figure 2. For RPV each RCE maintains a local list of active RV (Rendezvous View) and performs periodic exchanges of random RV to other RCE in its active RV. RCE also perform periodic purge of non-responsive RCE from its RV.

Copyright (c) IARIA, 2010 ISBN: 978-1-61208-105-2

E. Distributed Web services discovery

In our middleware each entity serves as a small scale UDDI using its LSR [4, 20]. Dynamic and partial view of network doesn't allow the leverage of a hash function for storing service descriptions and utilization of rendezvous mechanism between multiple registries [11] on ECE or RCE. Rather, we consider the following approach:

1. Upon receiving a query, entities check their LSC and recommend a target entity if a mapping of query with known expertise in semantic overlay is found.
2. The requesting entity can then take following actions:
   1) Re-query for a semantically similar expertise (if exact match is not available)
   2) Trigger of discovery protocol to query the services of recommended entity.
   3) Forwarding of request via RCE to other clusters, if request remains unanswered or is published again.

A detailed elaboration and rationale of all the above principles can be read in our previous work [20].

V. EXPERIMENTAL EVALUATION

The proposed middleware, in this paper, has been used in actual robot swarm [3]. Here in evaluation, we highlight the theoretical results using Player/Stage Simulator [10]. The communication infrastructure modeled for experiments is attributed by instability and partial communication coverage (attributed by robotic mobility). Web service descriptions are expressed in form of extended RDF triplets [4].

A. Discoverable services

Discoverable services refer to those services that have more than one point of discovery in the system. In other words, if a Web service is cached at another robot other than the service host/provider robot then that service is considered a discoverable service. Service discovery and invocation are considered different independent steps in service management and an additional host robot for Web services gives a higher probability of discovering a Web service when host robot is inaccessible (due to partial failures).

Measurement of discoverable services with respect to time measures the number of Web services that get cached at any other robot during the system execution. In other words, a higher number of discoverable Web services mean higher fault tolerance of system towards transient failures. Experimental result (Figure 3) shows discoverable services metric in the absence of conflict resolution. Experimental data is taken for varying Robot-RFID cluster ratios which is the number of robots sharing a MR/RFID.

Results reveal that the active communication mode achieves better performance in terms of discoverable Web services and with an increased Robot-RFID cluster ratio the performance deteriorates. This decrease in number of discoverable Web services, with higher Robot-RFID cluster ratio, is caused by increased number of conflict/messages-requests. This aspect will be shown in the comparison of active and passive communication mode

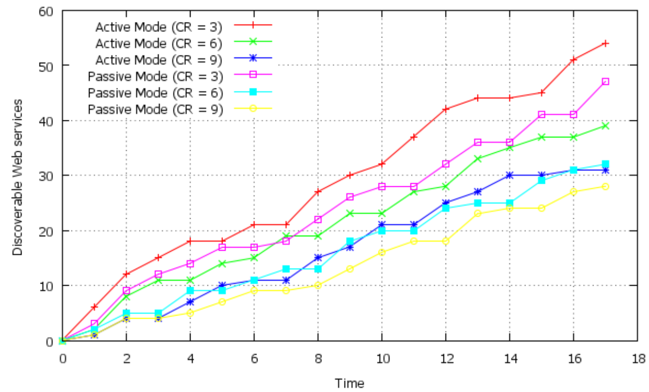

Figure 3. Discoverable Services without Conflict Resolution

Results reveal the effectiveness of active mode of communication over passive mode. This result gives a partial analysis, as we will observe latter that such improvement in number of discoverable Web services comes with an additional cost of increased bandwidth consumption in case of active communication and no real added value is achieved in terms of performance of actual Web services discovery.

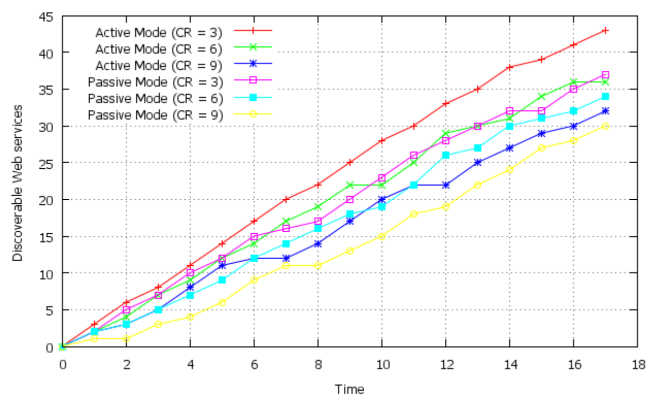

Figure 4. Discoverable Services with Conflict Resolution

Same experiment is performed with conflict resolution (Figure 4) as well. In this case the number of discoverable Web services with respect to time reveals a similar trend as that of non-conflict resolution (Figure 3). Curves of both active and passive mode (Figure 4) are mostly similar for various settings. Active communication mode still achieves better results as compared to passive communication mode i.e., more services are discoverable at any given time instant.

B. Impact of Cluster Ratio on Average Messages at a MR

In this experiment, we compare the impact of robot-RFID cluster ratio on the average number of messages disseminated in the environment (Figure 5). The result highlights a significant rise in average disseminated messages in active communication mode when robot-RFID cluster ratio is increased. On the other hand passive communication mode gives better results (i.e., fewer messages disseminated in the environment – less bandwidth intensive).

We can observe that, even with a higher robot-RFID cluster ratio, the rise in number of disseminated messages in passive communication mode is better to that of active communication mode. If RFID/MR memory size is increased, passive communication mode with a higher Robot-RFID cluster ratio as compared to that of active communication mode in same setting gives better results. This is shown in the overlap of "Active Mode – average msgs with RFID memory = 3" over "Passive Mode – Average msgs with RFID memory = 6". In other words, passive mode of communication is less bandwidth intensive even when a higher Robot-RFID cluster ratio is used.

Results reveal that passive communication mode exhibits controlled losses even with increased robot-RFID cluster ratio. With a higher RFID memory the results of passive communication mode improves further as compared to that of active communication mode. Active communication mode on the other hand incurs more losses with higher robot-RFID cluster ratios.

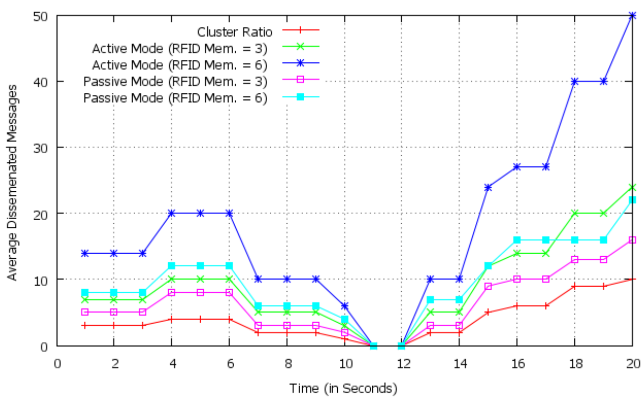

Figure 5. Impact of Cluster Ratio on Average Messages

C. *Message re-request overhead in Active vs. passive mode of communication*

Computation of number of message re-requests (see Figure 6 for non-conflict resolution mode) highlights the fact that, in general, there are fewer conflicts with smaller robot-RFID cluster Ratio. Passive communication mode serves better as it has less overhead (curves of passive mode with various settings show that numbers of conflicts are always lower than that of active mode of communication). In passive mode the conflicts are the number of request losses, while in active mode of communication conflicts are the service description losses. With a higher RFID memory size and lower robot-RFID cluster ratio both communication modes show similar results.

In case of conflict resolution mode (Figure 7) there is a significant difference in active and passive mode of communication. Such a difference is insignificant with higher robot-RFID cluster ratios. But with increased RFID memory size, passive mode of communication with conflict resolution gives improved results (lesser message re-request overhead). The curve for "Passive Mode – robot-RFID cluster ratio = 6" over "Active Mode – robot-RFID cluster ratio = 3" reveal this fact. The reason of such a behavior is the impact of service description memory requirement as compared to that of service requests in case of passive mode of communication. Such a behavior continues further with higher memory sizes of RFID/MR.

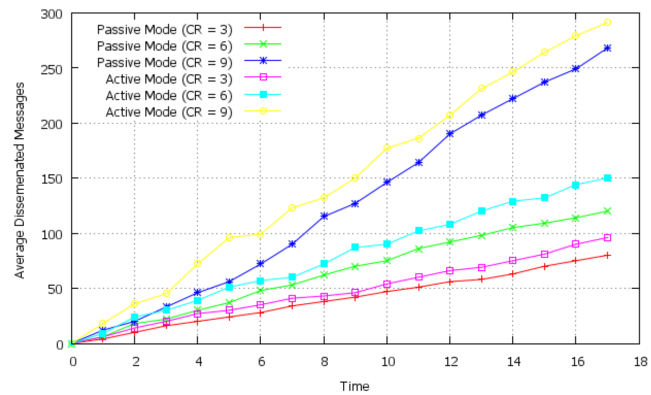

Figure 6. Active vs. Passive communication without CR

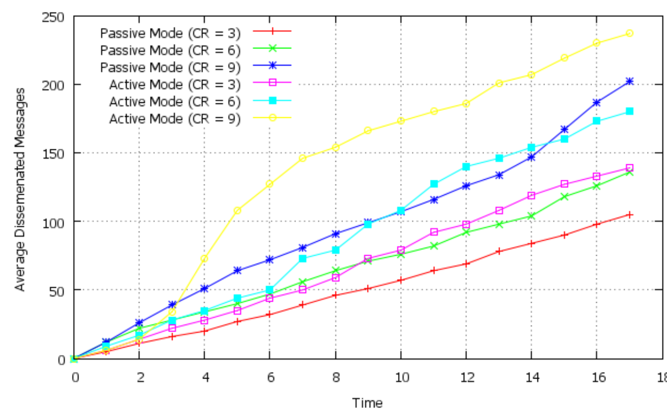

Figure 7. Active vs. Passive communication with CR

The comparisons (Figure 6 and 7) show that passive mode of communication serves much better in terms of number of conflicts and messages overhead as compared to active mode of communication. Passive mode shows better results with higher robot-RFID cluster ratio with higher memory size and with improved RFID memory it can support higher robot-RFID cluster ratios.

D. *Impact of caching and semantics in Service Discovery*

Lastly validation of the impact of caching and semantics in service discovery is done. Passive mode of communication is used with conflict resolution with different settings of robot-RFID cluster ratio. Results (see Figure 8) reveal that Web services discovery with caching augmented with semantics provides the best results (i.e., least number of hops/messages required for Web services discovery). There can be few abnormalities in result, for instance execution 14 shows better performance of syntactic discovery as compared to semantic service discovery with caching. This case represents a particular case exhibits a scenario in which a query is routed to a cluster (based on caching results) but due to mobility host robot has joined a different cluster – thus

Copyright (c) IARIA, 2010 ISBN: 978-1-61208-105-2

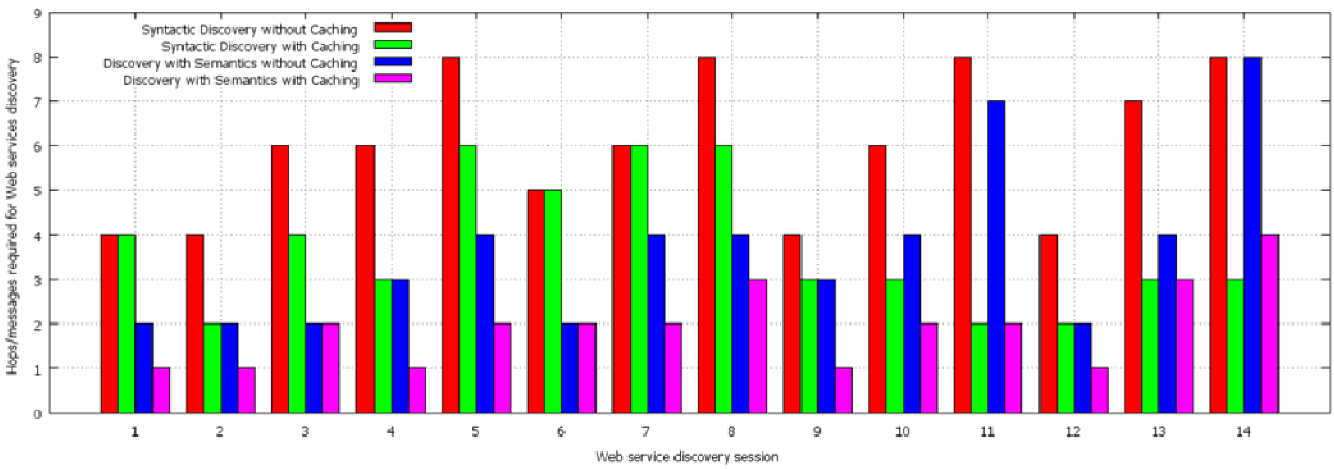

Figure 8. Impact of caching and semantics in service discovery

stale information in service cache effects additional number of hops across clusters for service discovery.

E. Overall analysis

Experimental results reveal the effectiveness of passive mode over active mode of communication. Active mode of communication (as employed in literature) gives result in number of discoverable services, however it does not fare well in terms of conflicts, higher robot-RFID cluster ratio and message re-request overhead. Moreover last result validates the improvements in service discovery and invocation of passive mode augmented with semantic discovery.

VI. Concluding Remarks

This paper gave a detailed evaluation of a passive distributed UDDI, a loosely coupled distributed UDDI solution for infrastructure-less communicating and heterogeneous systems. We proved, with the help of extensive experiments, that passive mode of communication performs considerably better than active mode of communication (i.e., existing solutions in literature) in terms of Web services discovery.

References

[1] Sun Microsystems Inc., “JXTA-SOAP bindings”.

[2] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, “Chord: Scalable P2P lookup service for internet applications”, SIGCOMM, ACM, 2001, pp 149-160.

[3] Knowledge Environment for Interacting Robot Swarms, http://www.roboswar.eu , (last accessed: August 31, 2010).

[4] A. Haseeb, M. Matskin & P. Küngas, “Light-Weight Decentralized Autonomic Web Service Discovery for Systems with Heterogeneous Communication Capabilities”, 12th IASTED IMSA’08, Hawaii.

[5] B. K. Kim, M. Miyazaki, K. Ohba, S. Hirai, and K. Tanie, “Web Services based robot control platform for ubiquitous functions”, Proc. IEEE Int. conf. on Robotics and Automation, , 2005, pp. 691- 696.

[6] Z. Wang and Y. Hu, “A P2P Network Based Architecture for Web Service”, Proc. IEEE Int. conf. on Wireless Communications, Networking and Mobile Computing, IEEE, 2007, pp. 3446-3449.

[7] S. Banerjee, S. Basu, S. Garg, S.J. Lee, P. Mullan, and P. Sharma, “Scalable Grid Service Discovery based on UDDI”, Proc. ACM 3rd Int. workshop on Middleware for grid computing, 2005, pp. 1-6.

[8] M. Cai and M. Frank, “RDFPeers: A Scalable Distributed RDF Repository Based on a Structured P2P Network”. Proc. ACM WWW conference, 2004, pp. 650 - 657.

[9] Z. Maamar, H. Yahyaoui, and Q. H. Mahmoud, “Dynamic management of UDDI registries in a wireless environment of web services: Concepts, architecture, operation, and deployment”. Journal of Intelligent Information Systems, Springer, 2007, pp. 105-131.

[10] B. P. Gerkey, R. T. Vaughan, and A. Howard, “The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems”. Proc. 11th Int. Conf. on Advanced Robotics (ICAR), Coimbra, 2003, pp. 317-323.

[11] Q. Lin, R. Rao, and M. Li, “DWSDM: A Web Services Discovery Mechanism Based on a Distributed Hash Table”. Proc. IEEE 5th Intl. conference on Grid and Cooperative Computing, 2006, pp 176-180.

[12] K. Arnold, B. Osullivan, R. W. Scheifler, J. Waldo, and A. Wollrath. “The Jini Specification (The Jini Technology)”. Addison-Wesley, Reading, MA, June 1999.

[13] The Salutation Consortium Inc 1999. Salutation Architecture Specification (Part 1), Version 2.1 Edition. http://www.salutation.org, last accessed: August 31, 2010.

[14] R. John, “UPnP, Jini and Salutaion - A Look at some popular Coordination Frameworks for Future Network Devices”. Technical report, California Software Labs, 1999.

[15] J. Veizades, E. Guttman, C. Perkins, and S. Kaplan, “RFC 2165: Service Location Protocol”, June 1997.

[16] S. Helal, N. Desai, and C. Lee. “Konark-A Service Discovery and Delivery Protocol for Ad-hoc Networks”. Proc. IEEE conf. onWireless Communication Networks, USA, 2003, pp: 2107 - 2113.

[17] D. Tang, C. Chang, K. Tanaka, and M. Baker. “Resource Discovery in Ad hoc Networks”. Technical report, Stanford University, August 1998. CSL-TR-98-769.

[18] D. Chakraborty, "Service Discovery and Composition in Pervasive Environments", PhD Thesis, June 2004.

[19] C. Pullela, L. Xu, D. Chakraborty, and A. Joshi. “Component based architecture for mobile information access”. Proc. Intl. Workshops on Parallel Processing, 2000, pp: 65 - 72.

[20] A. Haseeb, M. Matskin, and Peep Kungus, “Distributed Discovery and Invocation of Web Services in Infrastructure-less dynamic environments”. Pro. IEEE Intl. Journal of Web Services Practices (IJWSP), Vol. 4, 2009.

Copyright (c) IARIA, 2010 ISBN: 978-1-61208-105-2