

Scalability of Name Resolution for Ambient Networks

Pekka Pääkkönen¹, Nadeem Akhtar², Rui Campos³, Cornelia Kappler⁴, Petteri Pöyhönen⁵, Di Zhou^{6*}

¹ VTT Technical Research Centre of Finland, Kaitoväylä 1, 90571 Oulu, Finland

{Pekka.Paakkonen}@vtt.fi

² University of Surrey

Guildford, United Kingdom

{N.Akhtar}@surrey.ac.uk

³ INESC Porto

Porto, Portugal

{RCampos}@inescporto.pt

⁴ Siemens Communications

Berlin, Germany

{Cornelia.Kappler}@siemens.com

⁵ Nokia Research Center

Helsinki, Finland

{Petteri.Poyhonen}@nokia.com

⁶ PSE/Siemens

Vienna, Austria

{Di.Zhou}@siemens.com

Abstract. The convergence of mobile domain and data networks has been under focus in the standardization forums. However, dynamic interworking of wired infrastructure, wireless access systems and small scale Personal Area Networks has been challenging due to their heterogeneous nature. One of the most important problems to be solved is name resolution between different terminals and networks. This paper presents a new mechanism for name resolution, which relies on existing naming mechanisms. In particular the focus is on the scalability of the solution from signaling load and latency point of view.

1 Introduction

Ambient Networks (AN) is an integrated project of the Wireless World Initiative (WWI) financially supported by EU in its FP6 program. It presents a new networking concept enabling automatic, dynamic, flexible and on-demand cooperation of heterogeneous networks belonging to different operator and technology domains [1]. From a more pragmatic viewpoint this would mean the integration between devices, heterogeneous networks, and multiple accesses, which is a challenging goal due to

* The coauthors are ordered alphabetically.

geneous networks, and multiple accesses, which is a challenging goal due to the lack of a common control layer [1]. To overcome the problem the Ambient Control Space (ACS) concept has been introduced, which comprises of all the control functions (Connectivity, Mobility, QoS etc.) in a network domain and can be dynamically composed of several Ambient Networks to form a new AN [1]. To enable the cooperation between different ANs and connecting of functions between ACSs of the ANs, a Generic AN Signaling Protocol (GANS) stack has been developed [16]. GANS should support multiple access systems, user terminals and network domains. Therefore it is important that GANS is scalable for the heterogeneous environments. One of the most important properties of GANS is name resolution support for addressing and resolving endpoint(s) between the ACSs of the communicating ANs. GANS uses the services provided by a Destination Endpoint Exploration Protocol (DEEP) for name resolution, which is independent of the AN context. DEEP is a distributed name resolution protocol, which relies on existing naming mechanisms [17].

The most widely applied name resolution mechanism in the Internet infrastructure has been the Domain Name System (DNS), which enables resolution from a name to an IP address [9]. Measurements at two different locations in the Internet indicate that the median DNS latencies are below 100 ms, but the 90th percentile varies between 447 ms to 1176 ms [10]. Due to the problems of high latency, DNS server and gateway bottlenecks, implementation errors, misconfigurations, load imbalance, and update propagation a new name service for the Internet has been proposed [11]. It uses proactive caching with a Beehive replication framework, automatic load-balancing and fast propagation of updates, and should be compatible with legacy DNS and able to fully replace the existing DNS infrastructure.

The Session Initiation Protocol (SIP) [13] uses name resolution at the application level from a SIP URI to an IP address. SIP has been originally developed for the initiation and control of end-to-end sessions between SIP user agents, but in addition has other applications such as instant messaging and games [14]. SIP messages are transmitted to the destination via SIP servers, which are capable of resolving the SIP URI to an IP address of the endpoint. Usually DNS is used for finding the SIP server based on the SIP URI, but also local policies can be applied [13, 14].

Host Identity name space has been proposed to fill the gap between IP and DNS namespaces, to separate the IP layer from the transport, and to offer a basis for end-host mobility and multi-homing security problems [15]. The HIP name space consists of Host Identifiers (HI), which are used for identifying hosts. A HI is a public key of an asymmetric key-pair, and may be published with DNS or may be unpublished. Communication between HIP-nodes may be established peer-to-peer or via a Rendezvous-server [15].

Ambient Networks should be able to be applied in a wide context: in the fixed and mobile domain networks to Personal Area Networks, (PAN) which may not have access to the infrastructure. In this context DEEP is proposed as a generic name resolution mechanism, which relies on existing naming mechanisms, such as DNS, SIP and HIP, but is not targeted to replace them. DEEP is needed for several reasons: (1) Existing name resolution schemes have heterogeneous architectures and may be executed in different networking domains. DEEP is designed as a unified name resolution mechanism, which provides interworking between the existing name resolution

schemes and hides the heterogeneity related to them [17]. (2) Infrastructure support from DNS is not always possible in mobile access networks or ad hoc networks, and other solutions such as DHT [18] might need to be applied. This brings more complexity to the naming schemes, which should be able to interwork. (3) With the concept of moving networks in the AN context, the configuration of local name resolution information may need to be stored privately in an external name resolution system [17]. In this case updates to the global name resolution infrastructure (DNS) may not be needed when the local name-to-address mapping is changed. Thus it cannot be assumed that DNS can be applied in all the different networking environments of Ambient Networks. (4) Internet Multimedia Subsystem (IMS) is a signaling framework enabling end-to-end signaling between cellular terminals in heterogeneous networks. In the AN context the SIP based name resolution scheme of IMS cannot be seen to be sufficient (consider ad hoc networks as an example), given that it even has problems with interworking with the terminals in the WLAN domains [19].

The contribution of this paper is the presentation of DEEP in the AN context, and the scalability of DEEP. In particular DEEP has been analysed from the signaling load and name resolution latency point of view, which are important aspects of any protocol being developed for the Internet scale.

The results show the dependency of signaling load on the number of ANs and independency on the number of sessions in the AN context. In addition updating of sessions has been focused on. DEEP latency evaluation shows quite good scalability for name resolution delays smaller than 100 ms up to 10 intermediate DEEP-nodes. For bigger name resolution delays at the intermediate DEEP-nodes, the scalability of DEEP latency worsens.

The organization of the paper is as follows. An overview of DEEP name resolution design in the AN context is presented in Chapter 2, while Chapter 3 describes the implementation and simulation model used for the scalability study. The results, validation and evaluation of DEEP scalability are presented in Chapter 4. The conclusions made are stated in the last Chapter.

2 DEEP name resolution

DEEP enables symbolic name resolution to a locatable IP address of any remote Communication Endpoint (CEP). It is not dependent on the GANS protocol stack, and can be used outside the AN context. However, DEEP is presented with GANS, because it is important to understand that DEEP requires name resolution mechanism support from the framework it is being applied in (DNS, SIP etc.).

The structure of the Generic AN Signaling (GANS) protocol stack (Figure 2) follows the two-layer approach for generic signaling currently being defined by the NSIS working group in the IETF [5]. NSIS is a signaling protocol suite for manipulating control state in network elements on the data path of a flow. In the NSIS scope a signaling layer protocol (NSLP) is a generic term referring to any protocol within the application layer, which supports a specific signaling application [5]. The role of the NSIS Transport Layer Protocol (NTLP) is to transport signaling messages in the network independently of the particular NSLP [5].

The main differences between the GANS approach and the NSIS framework are in the addressing and routing of messages. In the NSIS scope it is assumed that the next NTLP signaling peer has knowledge of routing flow-based messages towards a destination, and a Router Alert Option [7] is used to enable NTLP-aware routers along a path to intercept packets. In GANS, the signaling is not always flow related, and the IP address of the signaling peer is not always known either. Rather, both symbolic names and IP addresses may be used for addressing the Communication Endpoint (CEP) in the destination AN. When the destination is addressed with a symbolic name, DEEP name resolution is performed. The Generic Internet Signaling Transport (GIST) as the instance of NTLP has been extended in order to support symbolic names in the Message Routing Information (MRI) [6], which is the main difference between GIST and EGIST. EGIST is a backwards compatible version of GIST, which means that GIST implementations in the network are able to interoperate with EGIST implementations despite of the modification. Generic Transport Layer Protocol (GTLT) refers to the combination of EGIST and DEEP protocols, which offers a transport service for GSLP (NTLP is the NSIS scope).

DEEP relies on sequential name resolution via one or more intermediate DEEP-nodes (Figure 2). The DEEP-nodes rely on existing local name resolution (NR) mechanisms (DNS etc.). DEEP is a stateless protocol, and follows the query-response messaging paradigm by using an EXPLORE message in the downstream and a RESPONSE in the upstream direction. DEEP can be run on top of any transport such as UDP or TCP.

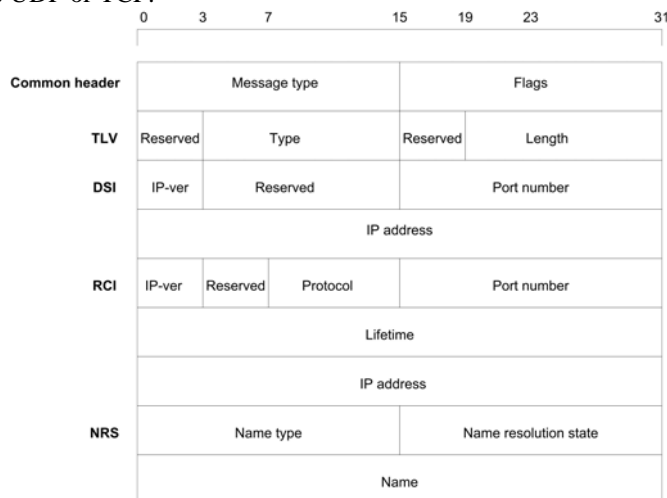


Fig. 1 DEEP message structure.

All DEEP messages have a common header and objects (DSI, RCI, NRS), each of which has a Type-Length-Value (TLV) header (see Figure 1) [17]. The common header expresses the type of the message, and has a flags field for extensions. Both EXPLORE and RESPONSE messages contain a Name Resolution State (NRS) object. NRS type indicates the syntax and semantics of the name, and NRS state ex-

presses the part(s) of the name, which have been resolved. EXPLORE has a DEEP Source Info (DSI) object, which includes the IP-version, port and IP address information of the source. RESPONSE has a Remote Contact Info (RCI) object, which includes the same information of the destination as the DSI object, but in addition lifetime of the name binding state is included. A binary format was selected for DEEP as opposed to a text-based format (e.g. XML) in order to minimize protocol overhead.

Figure 2 describes DEEP name resolution as part of GANS. GSLP at AN1 requests the sending of a message to AN2 with a symbolic name (CEP@AN2) (1). GSLP provides session-ID, GSLP-ID and MRI (symbolic name included) as parameters, which are used to identify a GTLP session and is saved by EGIST to the Message Routing State (MRS) table [6] (2). DEEP is used for name resolution, because a symbolic name is used for addressing the CEP (3). In this case only part of the name is resolved and a DEEP EXPLORE is sent towards the next intermediate DEEP-node (4). The intermediate DEEP-node uses its local name resolution service (5) in order to resolve the IP address of the gateway at AN2 and forwards the EXPLORE (6). The gateway in AN2 (gateway@AN2) receives the EXPLORE message and resolves the address of the final CEP, and forwards the message to it (7). The CEP (CEP@AN2) replies with a RESPONSE, which is sent back to AN1 (8). Finally AN1 receives the RESPONSE, and informs EGIST of a successful name resolution. EGIST adds a new entry to the MRS table, and sends further messages directly to the CEP in AN2.

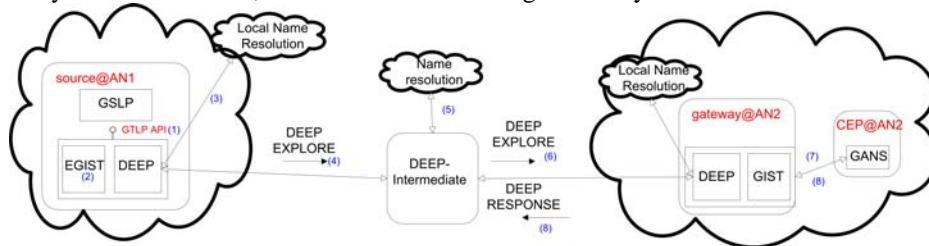


Fig. 2 An example of DEEP name resolution.

The mapping between a symbolic name and the corresponding IP address is called name binding state, which is included inside every MRI-object in the entries representing the GTLP sessions in the MRS table. The mapping may have to be updated in some circumstances, for example, because GSLP is willing to extend the lifetime of the binding. The updating may involve the execution of DEEP, which depends on local configuration or policies to be deployed in the AN context, both of which may vary. For example the destination DEEP-node may only be authorized to confirm the length of the lifetime to be extended.

3 Simulation model

The experiments performed in the simulation model tries to follow the principles discussed in [3] and [4] by varying a number of scalability variables in the network model and keeping the invariants unchanged. In the output other parameters are measured, based on which conclusions can be made.

The simulation model was synthesized in Network Simulator NS-2 [8] for the validation purposes of signaling load models in name resolution and update procedures. In addition the contribution of the procedures to the total signaling load was under study. The rationale and model for DEEP latency evaluation has been described in section 4.5.1. The GTLP protocol stack described in Figure 1 was implemented and consisted of the EGIST and DEEP protocols, which were executed on top of UDP. In addition a GSLP application and the required Tcl scripts were implemented in order to generate input over the GTLP API for the simulation scenario.

The simulation setup has been described in Figure 3. In the beginning of the scenario a different number of ANs perform name resolution and establish GTLP sessions with the destination AN (ANr) randomly within a 10 second time interval. This leads to the exchange of EXPLORE/RESPONSE messages (Name Resolution), and the sending of EGIST Data messages to AN-R (GTLP session setup). Finally each of the GTLP sessions is updated during a time period of 2 seconds at a random interval. ANi acted as a forwarder of traffic in the simulation setup.

The number of ANs (10,100,300), number of GTLP sessions (10,30,100) and probability of using DEEP in the update procedure (0;0,5;1) were used as input variables and simulations were executed with all combinations of them (3^3 cases). The above-mentioned variables were selected, because they were most obviously affecting the scalability aspects of DEEP in the hypothesized models. The number and sizes of DEEP messages were measured during the execution of the cases.

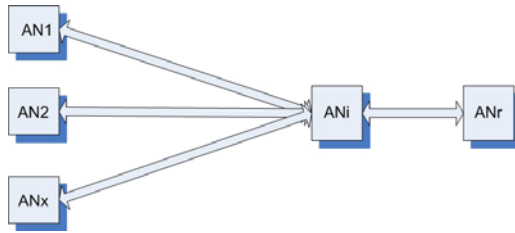


Fig. 3. The simulation setup.

4 Evaluation of results

4.1 Name resolution signaling load

The signaling load as bytes on the network from the point of view of an AN performing name resolution is given by equation (1). numberOfCEPs describes the number of CEPs in the destination AN identified by a different symbolic name. E and R express the size of EXPLORE and RESPONSE messages. The total signaling load as bytes on the network is given by equation (2), which is the sum of signaling loads of the

individual ANs performing the name resolution process (K = number of ANs). The model is applicable for a name binding lifetime.

The model described with equations (1) and (2) was validated with the name resolution phase in the simulation scenario, and show that the signaling load on the network is dependent on the number of ANs and doesn't depend on the number of sessions, when multiple ANs perform name resolution to an AN with a single CEP. Further results may be derived from the model by assuming that the overhead increases linearly to the number of CEPs to be contacted, since DEEP messages have to be exchanged when contacting the CEPs with different symbolic names. When a different number of destination ANs are considered, this would increase the load as a multiple of the CEPs and ANs to be contacted.

$$NR_signaling_load_AN = \sum_{i=1}^{numberOfANs} numberOfCEPs_i * (E + R) . \quad (1)$$

$$NR_signaling_load_total = \sum_{j=1}^K NR_signaling_load_AN(j) . \quad (2)$$

4.2 Update procedure signaling load

The updating of a GTLP session may involve the execution of DEEP as was explained in Chapter 2. Equation (3) describes the signaling load as bytes on the network, when an AN is performing the update procedure for a number of GTLP sessions. *numberOfSessions* describes the number of GTLP sessions to be updated with the remote AN, and P the probability of using DEEP in the update procedure or of updating a number of sessions with DEEP. A probability is taken into account, because DEEP may be used with the update procedure. Equation (4) describes signaling load as bytes on the network, which is a sum of the signaling loads produced by K ANs performing the update procedure.

$$UPD_sig_load_AN = \sum_{i=1}^{numberOfANs} numberOfCEPs_i * P * (E + R) * numberOfSessions . \quad (3)$$

$$UPD_sign_load_total = \sum_{j=1}^K UPD_sig_load_AN(j) . \quad (4)$$

The model was validated with the update phase in the simulation scenario by comparing it to the results generated with equations (3) and (4). Based on the validated equation it can be seen that signaling load on the network increases linearly with the number of GTLP sessions to be updated and the number of ANs performing the update. Further results can be derived from the model, when the number of ANs to be

communicated with is increased. In this case the signaling load should multiply as its function.

4.3 Total overhead

The total signaling load as bytes on the network is the sum of the signaling loads from the name resolution and update procedures and given by equation (5).

$$Tot = \sum_{i=1}^K NR_signaling_load_AN(i) + UPD_sig_load_AN(i) \quad (5)$$

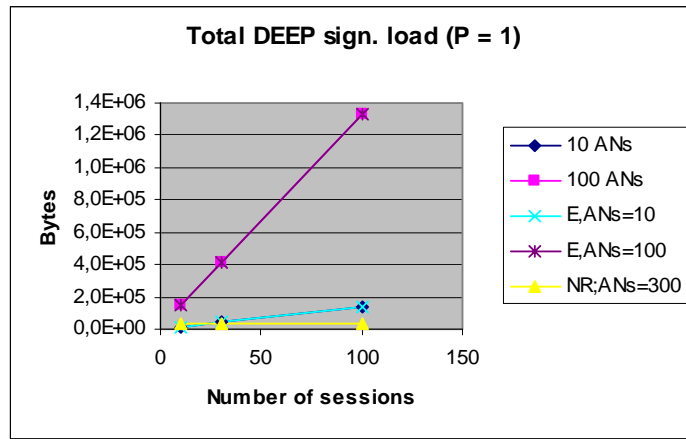


Fig. 4. Total *signaling load* on the network as measured in the simulation scenario and graphs drawn by the model (E=equation 5). In addition *signaling load* from the NR with 300 source ANs has been described.

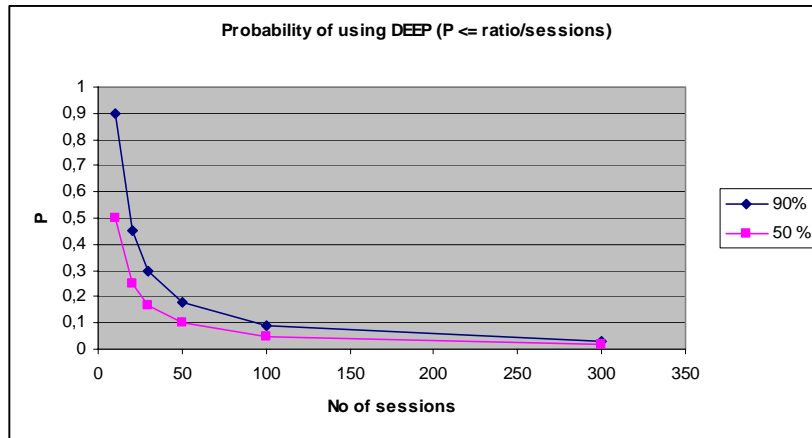


Fig. 5. The *probability of using DEEP* in the update procedure when the load from the update procedure is lower than 90% or 50% of the total signaling load.

The total signaling load on the network based on equation (5) was validated with the results measured in the simulation scenario. When comparing the load from the name resolution (Figure 4) to the total load, it can clearly be noticed that the load from the update procedure is dominating if each of the sessions are updated at least once during the lifetime of the name binding state. However it is not known at the moment how frequent the updates are or is DEEP involved in the update procedure. Because of this, further results on the probability of using DEEP in the update procedure have been described in Figure 5, which depicts the threshold for the probability of using DEEP in the update procedure in order to keep the share of signaling load from the update procedure lower than 90% or 50% of the total load. The equation in the figure has been derived from equations (2) and (4), when it is considered that an AN would perform name resolution for one AN and would execute the update procedure for a varying number of sessions.

4.4 DEEP message size

The size of DEEP messages is important from the signaling load point of view. The model (6) for DEEP message sizes as bytes were developed based on the message structure described in Figure 2.

$$\begin{aligned} \text{EXPLORE} &= 20 + \text{symbolicname} + \text{IPaddress} \\ \text{RESPONSE} &= 24 + \text{symbolicname} + \text{IPaddress} \end{aligned} \quad (6)$$

In the EXPLORE and RESPONSE messages the symbolic name and IP address consume most of the space in the message. If the symbolic name is assumed to be structured based on the Fully Qualified Domain Name (FQDN) [10], its maximum size is 255 bytes. An IPv4 address is 4 bytes and an IPv6 address 16 bytes long. In the simulations the size of the symbolic name was 100 bytes and an IPv4 address was used, which led to the following sizes: EXPLORE=124 bytes, RESPONSE=128 bytes.

4.5 DEEP latency

A number of factors must be taken into account, when studying the scalability of DEEP latency. There may be a different number of intermediate DEEP-nodes on the name resolution path from the source to the destination AN (Figure 1). The intermediate DEEP-nodes may or may not be on the shortest IP path between the peers, and there may be multiple IP hops/routers between the DEEP-nodes. The networks used between the DEEP nodes may be anything from wired high speed networks to wireless high-latency access systems. The name resolution infrastructure used by DEEP may be DNS or any other mechanism.

The model described by equation (7) was used, because in simulations a specific L1/L2 technology would have been selected. The simulations would have produced results only suitable for the selected L1/L2 technology. The goal was to determine scalability in terms of different RTTs between endpoints, and not focus on a particu-

lar lower layer. It has been assumed that DEEP name resolution is executed in the initiator and intermediate DEEP-nodes. The NR_delay describes the delay of the particular name resolution mechanism to be performed at intermediate DEEP-nodes, which has been assumed to be equal in each of the intermediate-nodes along the path. Interm_DEEP_nodes is the amount of intermediate DEEP-nodes on the name resolution path. RTT describes the time required for the EXPLORE/RESPONSE message exchange. Simulations were not executed for the latency evaluation, because RTT estimates would have become dependent on the L1/L2 solution, which was not the purpose of the study. The goal of the study is to explore the scalability of DEEP latency for networks with different RTTs, when the number of intermediate nodes and name resolution delay is varied. Figure 6 describes the latency of DEEP for a RTT of 300ms. The graphs with other RTTs follow a similar shape.

$$DEEP_lat = (Interm_DEEP_nodes + 1) * NR_delay + RTT . \quad (7)$$

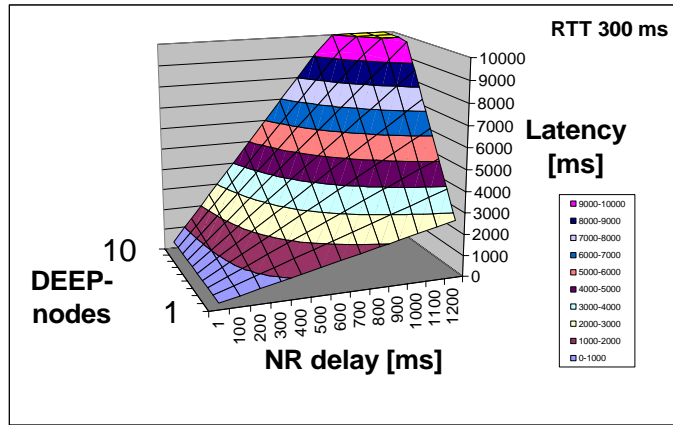


Fig. 6. DEEP latency as a function of the number of intermediate DEEP-nodes and NR delays when RTT of DEEP messages is 300 ms.

The results of DEEP latencies with RTTs up to 1500 ms, and their equations have been described in Table 1 (the equations have been derived based on (7) and the higher NR delay values in Table 1). DNS was considered as a name resolution mechanism in the study, and in particular the delays reported in [10] were used as a basis. Both median and 90th percentile of DNS latencies were considered and applied with model (7).

DEEP NR latency depends on the delay of the name resolution mechanism and topology of DEEP nodes in the Internet both of which are not known at the moment. Thus different RTT values, NR delays and number of intermediate nodes were explored with the proposed model. If DNS is considered to be used with DEEP as an existing name resolution mechanism supporting DEEP, and DEEP NR delay in intermediate nodes is assumed to be within the median DNS delays [10] (40-100 ms), the latency is expected to be under 2.6 seconds for all RTTs when intermediate

DEEP-nodes up to 10 are considered. In fact based on the equations it can be stated that the NR delay share is smaller than 60% of the total delay ($(10 \text{DEEP-nodes} * 100 \text{ms}) / 2600 \text{ms}$). However, when the typical median DNS delays [10] are considered, the results show that DEEP latencies up to 10 intermediate DEEP-nodes are significantly bigger (maximum between 13.5-57.6 seconds), because the NR delay share begins to dominate in the total latency.

Table 1. DEEP latencies and equations with different RTTs and number of intermediate DEEP-nodes.

RTT	300 ms	600 ms	1000 ms	1500 ms
NR delay				
40-100 ms (DNS median)	380-1400 ms (100*DEEP-nodes+400)	680-1700 ms (100*DEEP-nodes+700)	1080-2100 ms (100*DEEP-nodes+1100)	1580-2600 ms (100*DEEP-nodes+1600)
100-1200 ms (DNS 90th percen- tile)	500-13500 ms (1200*DEEP-nodes+1500)	800-13800 ms (1200*DEEP-nodes+1800)	1200-14200 ms (1200*DEEP-nodes+2200)	1700-14700 ms (1200*DEEP-nodes+2700)
1200-5000 ms	2700-56356 ms (5000*DEEP-nodes+5300)	3000-56656 ms (5000*DEEP-nodes+5600)	3400-57056 ms (5000*DEEP-nodes+6000)	3900-57556 ms (5000*DEEP-nodes+6500)

5 Conclusions

This paper presented the DEEP protocol for symbolic name resolution in Ambient Networks, and the results of a scalability analysis. In particular signaling load and latency aspects of the solution were focused on. The results show that DEEP name resolution signaling load on the network is directly proportional to the number of ANs and doesn't depend on the number of GTLP sessions. However, if multiple CEPs are to be contacted, then the load increases linearly, when contacting the CEPs with different symbolic names. Results in the DEEP update procedure show that signaling load on the network increases linearly with the number of GTLP sessions to be updated and the number of ANs performing the update. The overall DEEP signaling load on the network is the sum of the loads from the name resolution and update procedures. The results show that updating becomes a dominating factor in the total load, if the frequency of session updates or the number of sessions to be updated is high during the lifetime of the name binding state. It should however be noticed that the requirement for updating was derived from the AN context, and may not be needed in other frameworks. Models developed based on DEEP specification and implementation indicates that symbolic name and IP address form a major share of the DEEP message sizes, and determine the final size of the messages.

DEEP latency evaluation results show quite good scalability for name resolution delays smaller than 100 ms up to 10 intermediate nodes. In fact according to the previous studies, median DNS delays are lower than 100 ms. In this case the latency would be under 2.6 seconds for RTTs up to 1500 ms, and the share of the name resolution performed at the intermediate DEEP-nodes should be lower than 60% in the total latency. However if NR delays bigger than 100 ms are considered for RTTs up to 1500 ms, the results show that NR delays begin to dominate the total latency, which increases in direct proportion to the number of intermediate DEEP-nodes. Based on these results it can be stated that the scalability of DEEP depends in large part on the NR mechanism providing support for DEEP.

References

1. Niebert, N., et al. : Ambient Networks: An Architecture for Communication Networks Beyond 3G. *IEEE Wireless Communications*. 2 (2004) 14–22
2. Brunner, M.: Requirements for Signaling Protocols. IETF RFC 3726, (April 2004)
3. Floyd, S.,Paxson, V. : Difficulties in Simulating the Internet. *IEEE/ACM Transactions on Networking*. 4 (2001) 392–403
4. Floyd, S.,Kohler, E. : Internet Research Needs Better Models. *SIGCOM Computing Communications Review*. 1 (2003) 29–34
5. Hancock, R. et al.: Next Steps In Signaling (NSIS): Framework. IETF RFC 4080, (June 2005)
6. Schulzrinne, H., Hancock, R. : GIST General Internet Signaling Transport. IETF draft <draft-ietf-nsis-ntlp-08>, (September 2005)
7. Partridge, C. Jackson, A.: IPv6 Router Alert Option. IETF RFC 2711, (October 1999)
8. NS-2 website, URL: <http://www.isi.edu/nsnam/ns/>
9. Mockapetris, P. : DOMAIN NAMES-IMPLEMENTATION AND SPECIFICATION. IETF RFC 1035, (November 1987)
10. Jung, J., Sit, E., Balakrishnan H., Morris, R. : DNS Performance and the Effectiveness of Caching. *IEEE/ACM Transactions on Networking*. 5 (2002) 589–603
11. Ramasubramanian, V., Sirer, E.G. : The Design and Implementation of a Next Generation Name Service for the Internet. 2004 Conference on Applications, technologies, architectures, and protocols for computer communications. (2004) 331–342
12. Mockapetris, P.: Domain Names-Concepts and facilities. IETF RFC 1034, (November 1987)
13. Rosenberg, J et al.: SIP: Session Initiation Protocol. IETF RFC 3261, (June 2002)
14. Schulzrinne, H., Rosenberg, J. : The Session Initiation Protocol:Internet-Centric Signaling. *IEEE Communications Magazine*. 10 (2000) 134–141
15. Moskowitz, R., Nikander, P.: Host Identity Protocol Architecture IETF draft <draft-ietf-hip-arch-03>, (August 2005)
16. Akhtar, N.,et al. : GANS: A Signaling Framework for Dynamic Interworking between Heterogeneous Networks. submitted to VTC 2006
17. Pöyhönen, P.,et al. : DEEP - A Generic Name Resolution Protocol for Heterogeneous Networks. to appear in ICTTA'06
18. Araujo, F., et al. : CHR: a distributed hash table for wireless ad hoc networks. 25th IEEE International Conference on Distributed Computing Systems Workshops. (2005) 407-413
19. Marquez, F. G., et al. : Interworking of IP Multimedia Core Networks between 3GPP and WLAN. *IEEE Wireless Communications*. 3 (2005) 58–65