I-UDDI4M: IMPROVED UDDI4M PROTOCOL

Xiang Li[1], Lin Li[*], Lei Xue[1]

[1] *College of Information and Electrical Engineering, China Agricultural University, Beijing, China, 100083;*

[*] *Corresponding author, Address: P. O. Box 215, College of Information and Electrical Engineering University, China Agricultural University, 17 Qinghuadong Road, Beijing, 100083, P. R. China, Tel:+86-10-62732323, Email: lilincau@sohu.com*

Abstract   Web Service is growing to its peak for static networks, e. g. UDDI, but the improvement of the service for wireless networks, especially for MANETs is still required. At the same time, service discovery is basic technology in the MANETs. So, in this paper we propose an Improved UDDI Protocol used for MANETs (I-UDDI4m thereafter) in obtaining the information of Web Service actively, which is based on the theory of UDDI4m Protocol.

Key Words   I-UDDI4m; MANETs; UDDI; Web Services

1.   INTRODUCTION

Mobile Ad Hoc Network (MANET) is a temporary network that is constituted by some self-regulating wireless mobile nodes, does not have central controls and does not require any infrastructures. Because MANET does not have fixed routers, the nodes appear not only as terminal nodes which need information, but also as routers. Consequently, MANET is a self-organized, non-centre-controlled, multi-hop network system(Martin Mauve, 2001). The particularities of MANET are characterized by self-organization, highly dynamic topology, limited bandwidth, dynamic capability of the link layer, the restriction of the node caused by the equipment environment, multi-hop communication, distributed control, limited physical security, low expansibility of the network, half duplex wireless channel, and the short lifetime. More attentions are being paid to

MANET. and the discovery and description of Web Services will play an important role in the use of MANETs

Much work has been done on service discovery for MANETs recently, and several excellent protocols, such as Konark (Helal S, 2003), Allia (Ratsimor O, 2002) and MDFNSSDP (Gao Zhen-guo, 2006), were proposed. Cremonese *et al*. (Clement L, 2004) proposed the UDDI4m (UDDI for MANETs) protocol that is similar to but simplified Universal Description, Discovery and Integration protocol (UDDI) widely used in static networks. UDDI4m is a three-layer structure with the Application Layer, the Service Layer, and the Middleware Layer. The Service Layer is further partitioned into a UDDI4m API and an API for lower layer protocols. Therefore, the UDDI4m intensifies in expansibility and the ability of the protocol configuration.

Following the simplified UDDI entity construction proposed by Cremonese *et al*. (Clement L, 2004), we propose the Improved UDDI for MANETs protocol (I-UDDI4m) in which the mechanisms of dynamically Add operation and Delete operation are introduced into UDDI4m in this paper. Additional functions are expected to be included in the proposed protocol: writing available service information into UBRs by listening to the service query request in the network in order to avoid repetitious query,; and deleting useless information in the UBRs in time through listening to the changes of the topology of the network, which will lessen the amount of the useless query and reply, decrease the quantity of the data stream, save the bandwidth, and enhance the usability of the network

2.   THE ARCHITECTURE OF I-UDDI PROTOCOL

2.1   The Characteristics of MANET

MANET has its own characteristics compared with a static network as follows:

The topology of the network is changed rapidly and the transformation of the network is unpredictable.

MANET is based on the wireless links, so the bandwidth of a MANET network is narrower than that of the static networks.

The security is restricted by the wireless transformation.

The package lose-rate in MANET is higher, its latency is longer, and its shakiness is stronger compared with the static network,, and

MANET saves energy greatly.

Not all of the nodes in MANET have their own UBRs, the nodes which don't have UBRs (referred to Client Nodes) only own I-UIID4m clients. The

operations of service searching, publishing and deleting will be done by following procedures: the Client Nodes first find neighboring nodes which have UBRs (referred to Full Nodes), and then send requests to the Full Nodes. The Full Nodes will complete the operations they request.

2.2   The Characteristics of I-UDDI4m Protocol

For the design of I-UDDI4m, we should consider the Characteristics of MANETs. Service searching protocol in MANET must have some characteristics thereinafter:

The information of the UBRs must be updated in time in order to adapt the frequent changes of topology,.

Useless packages should be reduced in order to saving bandwidth.

All of the nodes are in the same order, a central control node is forbidden.

Useless information is deleted in time for the sake of reducing the energy consuming resulted by maintaining these information.

Based on these characteristics, I-UDDI4m is designed as a protocol with following specialties:

All of the nodes are in the same statuses, the central control node is forbidden.

For saving energy, not the whole web services information of MANET is kept in the UBRs.

The service request is monitored, the available service information in UBRs is updated in time, and the useless information is deleted based on the network state.

2.3   The Software Architecture of I-UDDI4m Protocol

Based on the analysis above, and following the architecture of the UDDI4m software, the software of I-UDDI4m protocol is divided into three layers: the application layer, the service layer, and the middleware layer. Among the three layers, the application layer is to provide a user interface, the middleware layer is to transform information and route. the service layer is to link the application layer and the middleware layer. Standard UDDI output messages are provided to the application layer, so that users can get information they need, while standard UDDI SOAP messages are enveloped for the purpose of the specific requirement of MANET and transmitted to the middleware layer. We provided a special listener, which is called Topology Changes and Services Available Listener (TCSAL for short), to monitor the topology changes and the status of web services, and update register information in UBRs. The software architecture of I-UDDI4m protocol is shown in Figure 1. We use Pastry protocol as the look up protocol in the

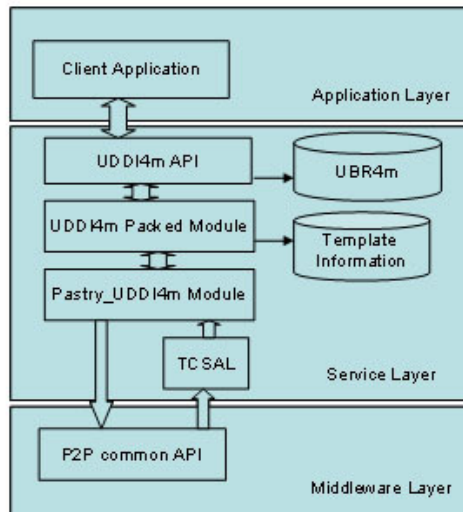middleware layer in order to keep it in consistent and comparable with UDDI4m

Fig. 1. the Software Architecture of I-UDDI4m Protocol

Cremonese and Vanni (Piergiorgio Cremonese, 2005) simplified the central entities of UDDI (shown in Figure 2), only including Business_Entity, Business_Service, Binding_Template, TModel and Contacts, and realized the main APIs of UDDI (Find_XXX(), Save_XXX(), Delete_XXX()). . We keep the same APIs as they are in standard UDDI.

The components of the Service Layer are as follows:

(1)   The UDDI4m API is to receiving the requests coming from the application layer and query information from local UBR4m (UBR for MANETs). If no information is found, the UDDI4m API would transfer the messages to the lower layer (only simplified entity is included), and process the messages which are sent back by the UDDI4m packed Module and in which the result of the query is included. Then the UDDI4m API updates the UBR4m, and sent the result back to the Application Layer.

(2)   The UDDI4m Packed Module is to accept the standard SOAP messages sent by UDDI4m API, and capsulate these messages with additional information (the node identifier and the life circle information) into a new package which is called Improved UDDI Package (IUP for short), and the information is written into the Template Information database. The structure of IUP is discussed in section three.

(3)   In order to be consistent with UDDI4m, we use Pastry protocol as the look up protocol in the middleware layer. The function of the Pastry_UDDI4m Module in our architecture is almost the same as the Pastry_UDDI4m Module in the UDDI4m architecture. It is used for pack up the data which are sent by the lower layer route protocol. It will be called by

TCSAL and sent the IUP which is independent from the router protocol back to UDDI4m Packed Module.

(4)   There is a semi-transparent listener which is called TCSAL under the Pastry_UDDI4m Module. It is used for monitoring the topology changes of the network and the responses of the service requests, It will match the information with the template information, call the upper layer API to execute, and then update the information in UBR4m.
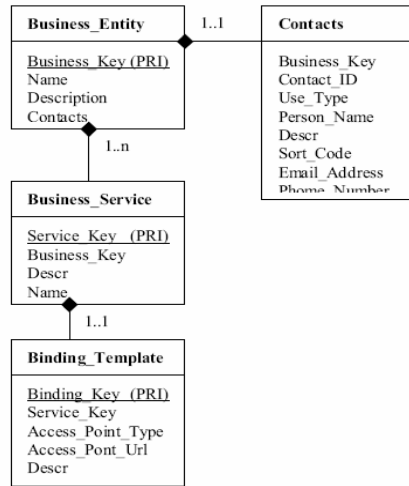
2.4   The Basic Data Package of I-UDDI4m

Fig. 2. A part of Relational Model UDDI4m registry

For achieving the specialty mentioned in section 2.2, the SOAP messages are packed up by the UDDI4m Packed Module before they are sent. The life circle information (such as life time, max hops) is added for the request messages, and the node identifier information is added to the response messages. The packages are shown in Figure 3.

Though some of the fields (such as the life circle information in Delete_XXX() and the node identifier information in Ret_Delete_XXX().. We will have a further discussion at section 3.3) are not in use yet because of its independence from the other nodes. We still keep these fields into the package for the purpose of keeping the ability of consistency and extension.
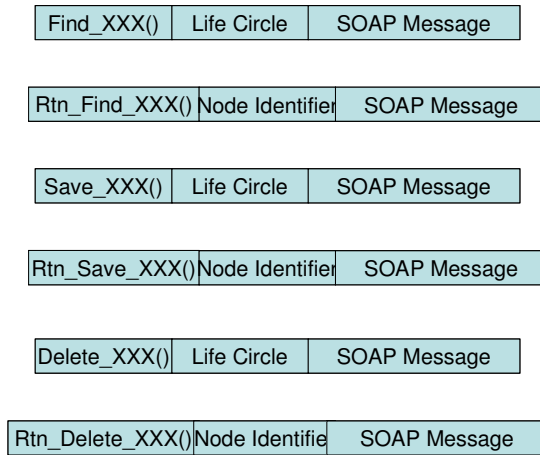
Find_XXX()	Life Circle	SOAP Message

Rtn_Find_XXX()	Node Identifier	SOAP Message

Save_XXX()	Life Circle	SOAP Message

Rtn_Save_XXX()	Node Identifier	SOAP Message

Delete_XXX()	Life Circle	SOAP Message

Rtn_Delete_XXX()	Node Identifie	SOAP Message

Fig. 3. The Structure of IUP Packages

3. THE CIRCULATE PROCESS OF I-UDDI4M

The primary operations for services which are included in MANET are listed here: Publish/Update Service information, Query Service according to some fixed characters, Delete Service, and Execute Service. We will discuss them respectively.
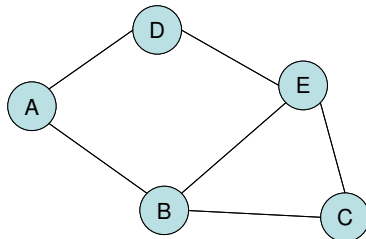

Fig. 4. A topology of a simple MANET

3.1 Publishing/Updating

The information is only needed to be written into local UBR4m when a service is published or updated.

3.2 Querying

The querying service is the key point which we pay special attention to, and a significant improvement is made here. When the query is send by node A in Figure 4, (a) firstly the query will fetch the UDDI4m API, and related

function will be called to query from the local UBR4m; (b) The fallback will be returned while the services are found; otherwise; (3) the query information will be packed up and sent to the middleware layer through the lower module (Pastry_UDDI4m Module in our paper); (4) The middleware layer will broadcast the message to its neighbour nodes. The neighbour node (take B for example) will submit the message to the upper layer when the node gets it. When TCSAL gets the message, it will be sent to the Pastry_UDDI4m Module; the Pastry_UDDI4m Module will unpack the message, and write the life circle information into the template information database, then execute the corresponding UDDI4m API. The UDDI4m API will query in its UBR4m, if the services are found, the result will be sent back, and (5) Otherwise, the process (4) will be continue till the max hops are exceeded or the life time is exhausted.

3.3   Deleting

The deleting operation is triggered by two different reasons, one is the node, the other is the TCSAL.

a) The deleting operation which is triggered by the node. While the node no longer provide the service, it execute the Delete_XXX() of the UDDI4m, and this will delete the information in local UBR4m.

b) The deleting operation which is triggered by the TCSAL. While the TCSAL monitored the topology is changed in the MANET, it will deliver the change to the upper layer, execute the appropriate UDDI4m API to finish the deleting. For example, when node C is not in the overlay of A, it cannot be fetched from A, the Delete_XXX() will be called and the service information which is correlative to C will be deleted.

3.4   Executing a Service

When a service is required, the node will firstly look up the local UBR4m. If the information of the service is found, the related service will execute, else, querying will be done.

If service is unavailable while it is called, TCSALs of the nodes in the path will receive this information, and each Delete_XXX() will be called to deleting information registered in the UBRs. In this way, the validity of the registered information is ensured, and the energy which is used to maintain UBRs is saved.

4. THE COMPARING WITH UDDI4M AND THE CONCLUSION

Comparing with UDDI4m, two aspects in I-UDDI4m are improved:

a) Service query is listened in order to updating the registered information in UBRs.

b) The topology changes are monitored for the sake of deleting the useless information in UBRs.

Due to the improvements mentioned on the above, I-UDDI4m, with the characteristics of UDDI4m, reduce the energy consuming, enhance the availability of the registered information; at the same time, it reduces the quantity of the data stream, saves the bandwidth, and the usability of MANETs is strengthened.

REFERENCES

Clement L, Hately A, Riegen CV, Rogers T. Universal description discovery & integration (UDDI) 3.0.2. 2004. http://uddi.org/pubs/uddi_v3.htm.

Gao Zhen-guo, Wang Ling, Zhao Yun-long, Cai Shao-bin Li Xiang. MDFNSSDP: A Minimum Dominating Forward Node Set Ba sed Service Discovery Protocol for MANETs. [J]ACTA ELECTRONICA SINICA. Vol. 34, No. 11. Nov.2006. 2030-2036.

Helal S, Desai N, Verma V, Lee C. Konark2a service discovery and delivery protocol for ad-hoc networks [A]. Proceedings of the Third IEEE Conference on Wireless Communication Networks [C]. New Orleans, USA, 2003. 2107 - 2133.

Martin Mauve, Jorg Widmer, Hannes Hartenstein. A Survey on Position-based Routing in Mobile Ad hoc Networks [J]. IEEE Network, 2001, 15 (6): 30-39.

Piergiorgio Cremonese, Veronica Vanni. UDDI4m: UDDI in Mobile Ad Hoc Network. [A]Proceedings of the Second Annual Conference on Wireless On-demand Network Systems and Services (WONS'05) [C]. Vol. 00. 26 – 31.

Ratsimor O, Chakarborty D, Joshi A, Finin T. Allia: alliance-based service discovery for ad-hoc environments [A]. Proceedings of the 2nd ACM International workshop on Mobile commerce [C]. Atlanta, Georgia, USA, 2002. 1 - 9.