VIP Bridge: Leading Ubiquitous Sensor Networks to the Next Generation

Shu Lei, Jinsung Cho, Soungyoung Lee

Department of Computer Engineering
Kyung Hee University, South Korea
{sl8132, sylee}@oslab.khu.ac.kr, chojs@khu.ac.kr

Abstract. Some applications need to collect information from several heterogeneous sensor networks which are physically deployed in different places to provide comprehensive services. In order to simplify the complexity of dealing with heterogeneous sensor networks, a uniform interface should be provided for users, which means that these several different sensor networks should be integrated over IP based wire/wireless networks into one virtual sensor networks to provide meaningful services for users. However, how to connect these heterogeneous sensor networks with IP based networks by a unique solo solution comes out to be an aforethought issue for this integration problem. In this paper, we first analyze and compare all the existing solutions for connecting sensor networks with IP based wire/wireless networks, then based on the analysis result we present the basic design principle and key idea of VIP Bridge for connecting sensor networks with IP based wire/wireless networks. By utilizing our proposed VIP Bridge, we describe a novel approach to integrate several different sensor networks into one virtual sensor network. Users can easily obtain data through both *Interest* based query and *IP address* based query from this virtual sensor network transparently. By having the comparison and prototyping work we claim that our new approach can cover most advantages of related researches. The most important contribution of VIP Bridge is that it first time successfully opens the door for leading ubiquitous sensor networks into the 4G paradigm [1].[1]

1 Introduction

"How can I manage so many sensor nodes which are deployed in 14 different buildings and using totally different routing protocols?"

- One day a university network administrator asked me

Wireless sensor networks are based on collaborative efforts of many small wireless sensor nodes, which are collectively able to form networks through which sensor information can be gathered. Such networks usually cannot operate in complete isolation, but must be connected to an external network through which monitoring and controlling entities can reach the sensor networks. As TCP/IP, the Internet pro-

[1] *** is the corresponding author.

tocol suite has become the de facto standard for large scale network, it is quite reasonable to connect wireless sensor networks with TCP/IP network to provide meaningful services for large number of Internet users. Furthermore, in the desired 4G paradigm [1], each mobile device will have global unique IPv6 address, all kinds of heterogeneous wireless networks and current existing IP based Internet should be integrated into one pervasive network to provide transparent pervasive accessibility and mobility for users. Internet users can seamlessly access and use services provided by heterogeneous wireless networks without knowing which kind of wireless networks they are. Sensor networks as a family member of wireless networks should also be integrated into this pervasive network and fully utilizes the advances of IPv6 address.

However, even though we know it is very important to connect sensor networks with TCP/IP network, the nature limitations of sensor networks, such as limited energy resource and low processing capability make it very difficult to deploy full IP protocol stack in sensor nodes. On the other hand, most of sensor networks are application-specific, which request different energy efficient routing protocols should be deployed for different corresponding application scenarios. In order to achieve the energy efficiency, routing protocols of different sensor networks should be chosen freely based on their application requirements. Therefore, for certain applications sometimes several different sensor networks which are locating in different places should be integrated into one virtual sensor network over the IP based wire/wireless networks to provide comprehensive services for remote users. These sensor networks may use totally different routing protocols to reduce the energy consumption for their application scenarios. How to easily and efficiently connect sensor networks with IP based wire/wireless networks, and finally enable all these sensor networks to be integrated into one virtual sensor network comes to be the critical research issue.

In this paper we propose a novel bridge based approach to integrate different sensor networks over IP based wire/wireless networks into one virtual sensor network. Through this virtual sensor network, user can easily query their interested information as well as directly query data from some special sensor nodes.

In next section, we present a short survey on related researches. Section 3 discusses the suitable communication paradigms of sensor networks for connecting with TCP/IP network. In section 4, we present the major principle of designing new solution. Section 5 presents the key idea and detailed description of our VIP Bridge. A case study is provided in section 6. In section 7, we present the comparison between related researches and our approach; in addition, we show that we can easily integrate several different sensor networks into one virtual sensor networks by using our VIP Bridge. In section 8, we present the initial implementation work. Finally, we conclude this paper in section 9.

2 Related Work

Since the attention of present research community is mostly paid to other issues of sensor networks, such as energy efficiency and security, very limited numbers of related researches have been performed. Basically, related researches can be categorized into two different approaches: 1) *Gateway-based approach*; 2) *Overlay-based approach*.

***Gateway-based approach*:** This is the common solution to integrate sensor networks with an external network by using *Application-level Gateways* [4] as the interface. The recently proposed IETF draft [5] and existing ZigBee Gateway [6] are also following this approach. Different protocols in both networks are translated in the

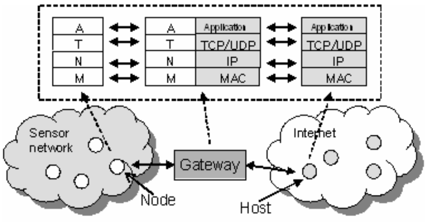


Fig. 1. Application-level Gateway

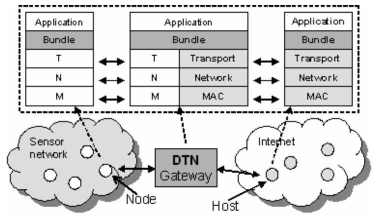
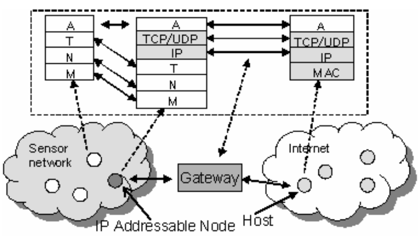


Fig. 2. Delay Tolerant Network

application layer as the Figure 1 shows. The main role of this gateway is to relay packets to different networks. The advantage is: the communication protocol used in the sensor networks may be chosen freely. However, the drawback is: Internet users cannot directly access any special sensor node by using current existing IP based working model. Another research work, *Delay Tolerant Network* [7], also follows this *Gateway-based approach*. The key different point from [4, 5, 6] is that a *Bundle Layer* is deployed in both TCP/IP network and non-TCP/IP network protocol stacks to store and forward packets, as Figure 2 shows. It is very easy to integrate with different heterogeneous wireless networks by deploying this *Bundler Layer* into their protocol stacks. But the drawback also comes from the deployment of *Bundle Layer* into existing protocols which is a costly job, and it cannot be expected that every deployed sensor node can has enough resource and processing capability to support this *Bundler Layer*.

***Overlay-based approach*:** There are two kinds of overlay-based approaches for connecting sensor networks with TCP/IP network: 1) *TCP/IP overlay sensor networks*; 2) *sensor networks overlay TCP/IP*. Research work in [8, 9] provides a solution to implement IP protocol stack on sensor nodes which is named as **u-IP**. The key advantage is: Internet host can directly send commands to some particular nodes in sensor networks via IP address. However, this **u-IP** can only be deployed on some sensor nodes which have enough processing capabilities. Another problem is that



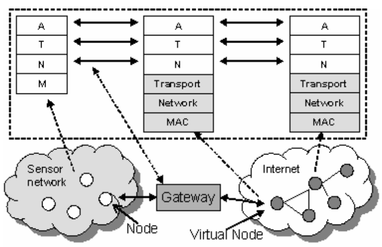


Fig. 3. TCP/IP overlay sensor networks

Fig. 4. Sensor networks overlay TCP/IP

the communication inside sensor networks based on IP address will bring more protocol overhead, e.g. tunneling. We show this **u-IP** approach in Figure 3. The *sensor networks overlay TCP/IP* is proposed in [10]. As Figure 4 shows, sensor networks protocol stack is deployed over the TCP/IP and each Internet host is considered as a virtual sensor node. By doing so, the Internet host can directly communicate with sensor node and Internet host will process packets exactly as sensor nodes do. The problem of [10] is: it has to deploy an additional protocol stack into the Internet host, which brings more protocol header overhead to TCP/IP network. In addition, it loses the consistency with current IP based working model, which makes it not suitable to meet requirements of Next Generation Network paradigm.

By analyzing related researches, it is not difficult to figure out that ***we must propose a new approach which can cover all advantages of existing researches and still have consistency with IP based working model to realize the Next Generation Network paradigm***. So, what are major principles for designing this new approach? Before presenting these major design principles let us have a look at the different communication paradigms of sensor networks for more detailed analysis.

3 Communication Paradigms of Sensor Networks

Typically, there are three kinds of communication paradigms in sensor networks: 1) *Node-Centric*, sensor nodes are labeled with some IDs and routing is performed based on these IDs, e.g. some table-driven-routing protocols; 2) *Data-Centric*, trying to make sensor networks answer "Give me the data that satisfies a certain condition", e.g. Directed-Diffusion [11]; 3) *Location-Centric*, using the location of sensor nodes as a primary means of address and routing packets, e.g. CODE [12].

Then, which communication paradigm is suitable for connecting sensor networks with TCP/IP network? In nowadays Internet, every network entity such as personal computer, router, or printer has its own IP address for identifying itself from others. Commercial databases used to provide diverse services for Internet users are stored in different computers. Internet users can access these services by using the IP addresses of those computers. However, the difficulty of remembering IP address for service motivates the using of Domain Name, which probably uses the name of this service. Internet users can easily use the Domain Name to access the corresponding service, with the assumption that this service's domain name or IP address can be known by users in advance. The routing in Internet is also IP address based. ***This kind of working model is similar with those of Node-Centric and Location-Centric.***

Data-Centric approach presented in paper Directed Diffusion [11] has its foremost different assumption from the IP based Internet working model: ***users don't know the exact locations of their interested sensors or data in advance***. In order to find the needed data, users request the gateway to broadcast the *Interest packet* to all the sensor nodes of sensor networks and look for the data source. On the other side, the sensor nodes which have the needed data also broadcast the *advertisement packet* to tell other nodes that they have this kind of data. Once the *Interest packet* and *advertisement packet* meet each other in certain sensor node, the transmission path from data source to gateway will be set up. If we consider the data provided by these sensor nodes as the services, we realize that the working approach of *Data-Centric* is more like a Service (Data) Discovery approach, and the sensor networks is considered as a kind of database system.

Now we can easily answer that "***In order to provide the consistency between the working models of sensor networks and TCP/IP network, the Node-Centric and Location-Centric communication paradigms are more suitable for connecting sensor networks with TCP/IP network."***

After having these aforementioned analyses, we can present our major design principles in the following section now.

4 Major Design Principles

These following principles of designing our new approach in some sense are also our final design goals which must be clearly figured out, so that we can successfully deploy a comprehensive approach to connect heterogeneous sensor networks with IP based wire/wireless networks as well as integrate them into one IP address based virtual network towards to the Next Generation Network paradigm.

Consistency: Either the current existing internet or the future coming Next Generation Network is actually having the working model based on IP addresses. If ubiquitous sensor network wants to be successfully integrated into the pervasive network, ***it should be IPv6 address based, so that it can have the consistency with the working paradigm of Next Generation Network***.

Transparency: Transparency is a key feature and basic requirement for pervasive network. Non-system-designer users should be able to use services provided by underlying IP address based networks without knowing that what kind of networks actually provided these services. Even though ubiquitous sensor networks have a lot of distinctiveness from other IP based networks, however in terms of providing services to end users, they should follow the same principle which means ***non-system-designer users should be able to use service provided by sensor networks without knowing that "these services are provided by certain sensor networks"***.

Direct accessibility: In IP based wire/wireless networks, end users are able to directly access some service by using IP address, for example by using 'http://163.180.140.34' to access VIP Bridge website. Similarly, some applications of sensor networks also request that some sensor nodes should be able to be directly accessed and operated by end users. Most of these traditional approaches only can accomplish this direct accessing by using sensor node's ID or geographic location address, since they actually do not using IP addresses for routing. However, in order to keep the above mentioned Consistency and Transparency, we must ***let the end users to be able to directly access and operate some sensor nodes by using IP addresses, which means every sensor node should be able to use IP address to identify itself from others globally***.

Energy efficiency: Ubiquitous sensor networks are normally battery equipped which naturally request the usage should be energy efficient. Most of sensor networks are application-specific, which request different energy efficient routing protocols should be deployed for different corresponding application scenarios. In order to achieve the energy efficiency, ***routing protocols of different sensor networks should be chosen freely based on their application requirements***, which means within one virtual sensor network several different sensor networks may respectively use totally different routing protocols.

No overlay approach: From the related research work, we can easily observe that either the *TCP/IP overlay sensor networks* or *sensor networks overlay TCP/IP* require modification on protocol stacks. Both approaches will bring extra overhead to either the sensor networks or IP based wire/wireless networks. More important thing is that ***both approaches have very strict constraints for realistic deployment in terms of the integration of heterogeneous sensor networks***. For *TCP/IP overlay sensor networks*, we cannot expect all these different sensor networks can have enough processing capacity to embed the IP stack inside, and for *sensor networks overlay TCP/IP* we cannot expect all these different sensor networks are fortunately use the same kind of routing protocol.

Easy integration: In the coming Next Generation Network, different IP based wire/wireless networks can be easily integrated by using IPv6 addresses. A common IP layer is used in their protocol stacks. ***In order to easily integrate ubiquitous sensor networks with other IP based wire/wireless networks, we should also deploy IP addresses for sensor networks***. Once we can make several different sensor networks be equipped with IP addresses, we can easily integrate them into one virtual sensor network.

Plug and Play: Sometimes, in order to provide additional information for end users or extend the functionality of whole integrated virtual sensor network, some new sensor networks will be deployed into some new places while using some new routing protocols. If routing protocols used by these new sensor networks are fortunately sensor node ID based or geographic location address based, then we can easily assign IPv6 addresses to them. ***These newly deployed sensor networks will be able to be integrated into the virtual sensor network easily by using VIP Bridge***.

Taking the advantage of sensor node's unique information: IP address is generally unique globally. In order to let each sensor node be globally unique and equipped with IP address, we should take the advantage of sensor node's unique information. As we presented in section 2 many sensor networks are *Node-Centric* or *Location-Centric*, both sensor nodes' label (ID) and location addresses are unique information inside sensor networks, it can be used to identify each sensor node form others. ***By using this unique information, we can easily set up the corresponding mapping between IP addresses and sensor nodes' IDs or location addresses***.

5 VIP Bridge

5.1 Key Idea

Taking all of these foregoing principles into consideration, we create our key idea ***VIP Bridge: Basing on Node-Centric or Location-Centric communication paradigm, mapping the node label (ID) or location address with IP address in bridge. The IP address will not be physically deployed on sensor node, but just store in bridge as a virtual IP address for Internet users. Each of these assigned virtual IP addresses is globally unique. Packets that come from one side will be translated into corresponding packet formats and sent to another side by this VIP Bridge. The packet routing is based on these virtual IP addresses and users' original IP addresses.***

5.2 Overview of VIP Bridge

The component based overview of VIP Bridge is showed as Figure 5. Two Packet Analyzers analyze packets from both application layer and sensor network layer respectively, and two packet translators translate packets for both sides: *1) TCP/IP Network -> Sensor Networks (T->S) Packet Translator,* translating packets from TCP/IP network into the packet format of sensor networks; *2) Sensor Networks -> TCP/IP Network (S->T) Packet Translator*, translating packets from sensor networks into the packet format of TCP/IP network. We use *T->S Packet* to represent the packet that comes from TCP/IP network, and *S->T Packet* to represent the packet that comes from sensor networks. A *Node ID/Location Address* is the node ID or location address of a sensor node. A *Data Information* is a description about what kind of data can be provided by this sensor node. An *IPv6 Address* is the assigned IP address for this special sensor node.

VIP Bridge will actively discover sensor nodes and ask them to register *Data Information*, *Node ID/Location Address,* and also actively assign *IPv6 Address* for these sensor nodes. All these information are stored in a Repository which physically locating in the VIP Bridge. Furthermore, ***bridge will map these three different kinds of information with each other.***

After packet analysis, query packets are sent to Query Engine to extract the necessary information from Repository to compose the new packet format. The Mapping

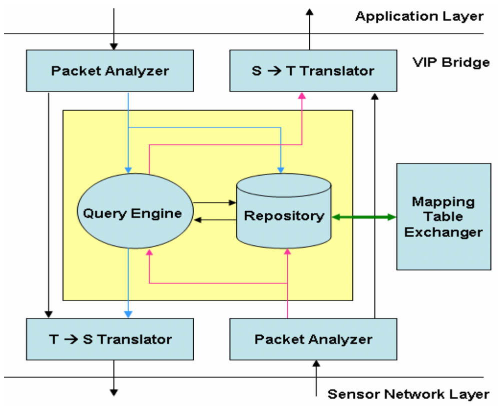


Fig. 5. Component based overview of VIP Bridge

Table Exchanger exchanges *Mapping Tables* between different VIP Bridge*s*, in order to integrate all *Mapping Tables* into one. In our approach, we use XML to express our *Mapping Tables*. SOAP (**S**imple **O**bject **A**ccess **P**rotocol) is a lightweight XML-based messaging protocol used to encode the information in Web Service request and

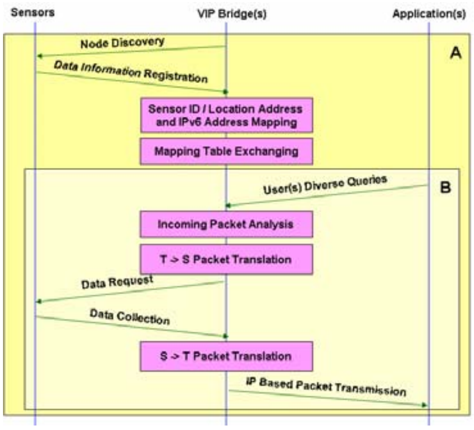


Fig. 6. High Level Overview of VIP Bridge Operation

response messages before sending them over a network. SOAP messages are independent of any operating system or protocol. The SOAP protocol extends XML so that computer programs can easily pass parameters to server applications and then receive and understand the returned semi-structured XML data document. In our approach, all the communication between different VIP Bridges is accomplished by SOAP.

The high level overview of VIP Bridge operation is show in Figure 6. After node discovering, sensor nodes register their *Node ID/Location Address* and *Data Information* into VIP Bridge. VIP Bridge should create the *Mapping Table* and assign the *IPv6 Address*. Before receiving queries from users, VIP Bridges should exchange

Mapping Tables to create the view of virtual sensor networks. After all these preparation in part A, the real query process can start as in part B.

The incoming queries will be analyzed to classify them into two types of queries, as Figure 7 shows: 1) IPv6 address based query, which allow users to directly access some special sensor nodes; 2) User's *Interest* based query, which is used to support the standard query of sensor networks, such as Direct Diffusion. Because sensor nodes have already register their *Data Information* in VIP Bridge in advance, and there are mapping tables between *Data Information* and *IP Address* in VIP Bridge, it is very easy to search the corresponding IP addresses based on user *Interest*.

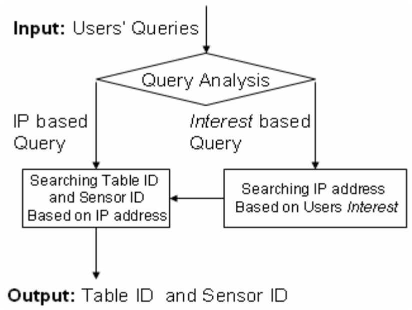


Fig. 7. Query Process

5.3 Packet Format Description

The packet format of original *T->S Packet* has four major fields, as showed in Figure 8:

1) *User IP*, used to represent the IP address of user who sends this packet;
2) *Sensor IP/Bridge IP*, used to represent the destination of this packet, which can be the bridge IP address or some special sensor node's IP address;
3) *Q/O*, used to represent packet type: *Query Command* or *Operation Command*;
4) *Complicated/Simple Data Request / Operation Command*, used to represent the real content that is carried by this packet.

The packet format of created *T->S Packet* has the following four major fields:

1) *Bridge ID/Location*, used to represent the ID or location address of Bridge, which sends the packet to sensor networks;
2) *Sensor ID/Location*, used to represent the ID or location address of data source;
3) *Q/O*, used to represent packet type: *Query Command* or *Operation Command*;
4) *Complicated/Simple Data Request / Operation Command*, used to represent the real content that is carried by this packet.

The *Query Command* is used to request data from sensor networks, it can be as simple as query data just from one special sensor node, or it can be as complicated as query data from many sensor nodes at the same time. *Operation Command* is used to remote control one special sensor node's working status.

Similarly, the packet format of *S->T Packet* also has four major fields:

1) *Sensor ID/Location*, used to represent the ID or location address of data source;

2) *Bridge ID/Location*, used to represent the ID or location address of Bridge, which is the destination of this packet;

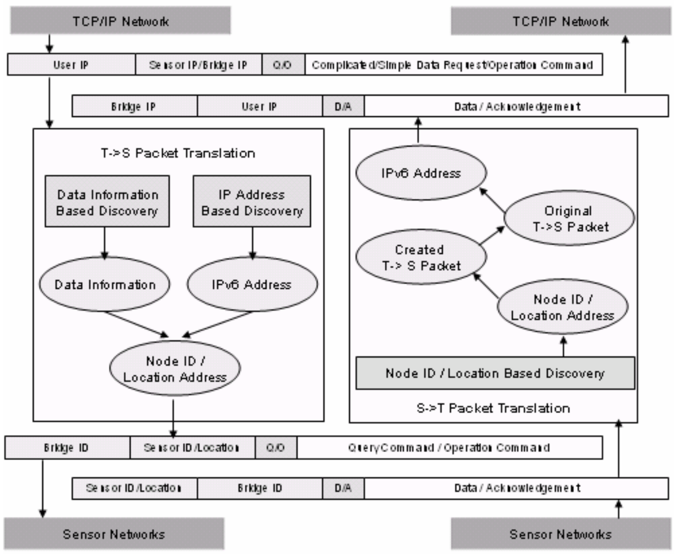


Fig. 8. Packet formats and Packet translators of VIP Bridge

3) *D/A*, used to represent packet type: *Data Packet* or *Acknowledgement Packet*;
4) *Data/Acknowledgement*, used to represent real content carried by this packet.

The packet format of created *S->T Packet* has the following four major fields:

1) *Bridge IP*, used to represent the IP address of Bridge, which sends the packet to TCP/IP network;
2) *User IP*, used to represent the IP address of receiver;
3) *D/A*, used to represent packet type: *Data Packet* or *Acknowledgement Packet*;
4) *Data/Acknowledgement*, used to represent real content carried by this packet.

The *Data Packet* corresponds to the *Query Command*, and the *Acknowledgement Packet* corresponds to the *Operation Command*.

5.4 Workflow of Both Translation Components

In this subsection, we will present the detailed workflow of two translation components to explain how we translate different packets for both sides.

TCP/IP Network -> Sensor Networks Packet Translation: After receiving packets from IP based wire/wireless network, there are two ways to translate them into the packet format that used by sensor networks: 1) *Data Information Based Discovery*; 2) *IPv6 Address Based Discovery* as showed in Figure 8. The translation workflow is showed in Figure 9.

Bridge will analyze these received packets based on the field "*Q/O*" to categorize them into *Query Command* and *Operation Command*. If a packet is an *Operation*

Command, then bridge can base on the *Sensor IP* to search the Repository to find out the corresponding *Node ID/Location Address* of this sensor node through the mapping between *IPv6 Address* and *Node ID/Location Address*. If a packet is a *Query Command*, then bridge can base on *Complicated/Simple Data Request* to search the database to find out the corresponding *Node ID/Location Address* of this sensor node through the mapping between *Data Information* and *Node ID/Location Address*. After knowing *Node ID/Location Address* of this sensor node, we can easily create the new packet for sensor networks. Before sending new created packet to sensor networks, we backup this new *T->S packet*, and map it with the original *T->S packet* in bridge. These saved packets will be used when we translate packets that come from sensor networks into the packet format of TCP/IP network.

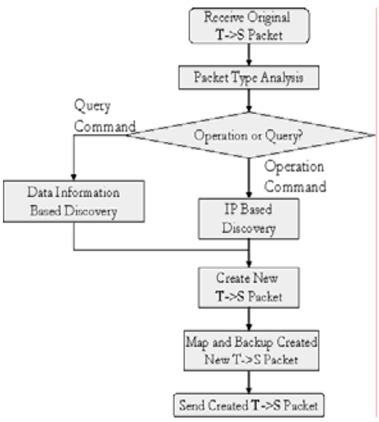


Fig. 9. Translation workflow of T->S

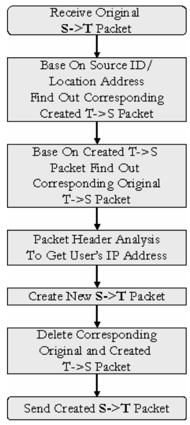


Fig. 10. Translation workflow of S->T

Sensor Networks -> TCP/IP Network Packet Translation: The workflow of *S->T translation* is showed in Figure 10.

After receiving the *S->T Packet* from sensor networks, bridge first bases on packet's *Sensor ID/Location* to find out the created *T->S Packet*, then through the mapping between the created *T->S Packet* and the original *T->S Packet*, bridge can easily find out the original *T->S Packet*.

By analyzing the original *T->S Packet*, bridge can get the *User IP*, and then create the new *S->T Packet*. Before sending this new *S->T Packet*, bridge will delete the corresponding original and created *T->S Packets* to save the storage space of the database.

6 A Case Study: Z-IP Approach

In this section, we present a case study named Z-IP approach to show that how we can connect one sensor node ID based sensor network with IPv6 based wire/wireless

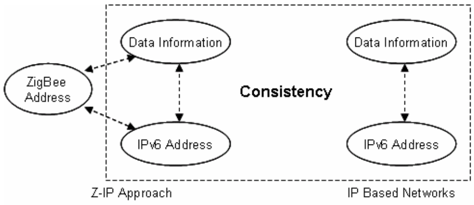


Fig. 11. Hide the ZigBee Address to make the consistency with TCP/IP network (Internet)

networks. We use the existing famous ZigBee [13] as the routing protocol in this *Z-IP approach*. In ZigBee based sensor networks, every sensor node has its own unique

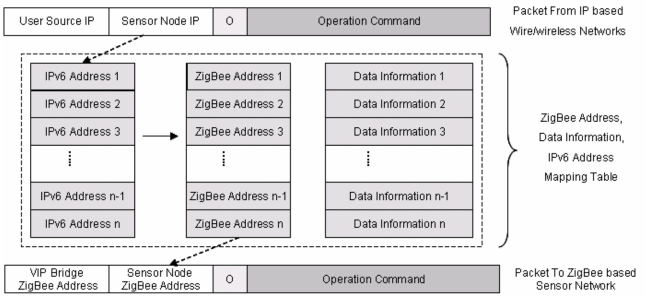


Fig. 12. Directly operate sensor node

ZigBee address as the sensor node's ID. After building up ZigBee based sensor networks, each sensor node actively senses its local environment and registers the *Data*

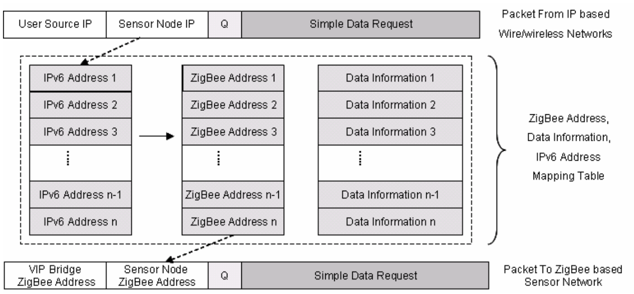


Fig. 13. Directly Query based on IPv6 address

Information about the sensed data to VIP Bridge. By doing so, VIP Bridge can have *Data Information* and ZigBee addresses for whole sensor network in advance. Sequentially, VIP Bridge assigns global unique IPv6 address for each ZigBee address in Repository. Technically, it is possible to assign IPv6 address to every sensor node because IPv6 can provide enough IP address for whole sensor networks. Furthermore,

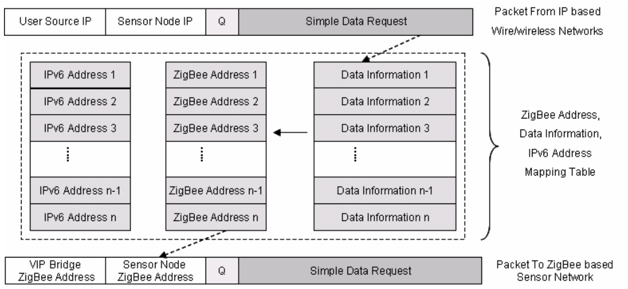


Fig. 14. Directly Query based on Data Information

we make the Data Information, ZigBee address, IPv6 address mapping in VIP Bridge as the left part of Figure 11. By doing this kind of mapping, whenever IP based

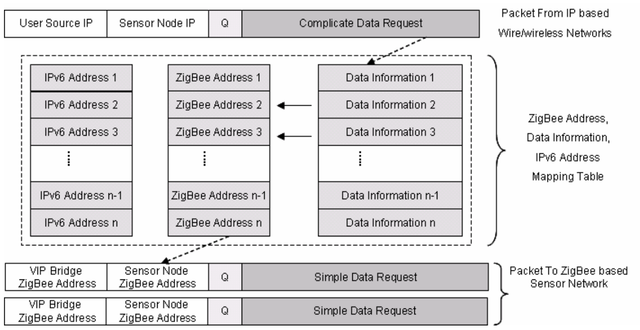


Fig. 15. Complicated Data Request from several sensor nodes

wire/wireless network users want to get some data, they can easily find out the exact sensor node through the corresponding IP address and ZigBee address. However, we

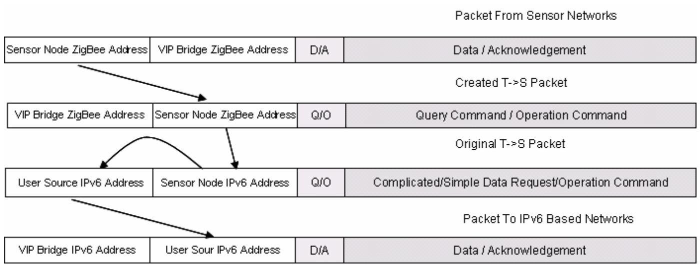


Fig. 16. Send *S->T Packet* to TCP/IP network

choose to use IPv6 address instead of ZigBee address which means we hide the ZigBee address but only release the IPv6 address to some authorized IP based network users through the website, By adopting this changing we can achieve the consistency between traditional IPv6 based Internet and our *Z-IP approach* as well as the transparent direct accessibility, as Figure 11 shows.

In order to clearly explain the packet translations inside VIP Bridge, we present several different kinds packet translations here as examples. In Figure 12, one IP based network user sends an Operation Command to one special sensor node to change its working status. The targeted IP address is the IPv6 address of this special sensor node. Here, we assume that this IPv6 network based no-system-designer user can get this targeted IPv6 address from some public website which released by sensor networks developer. After receiving the packet from IPv6 based network, the VIP Bridge will search the mapping table to find out the ZigBee address of this sensor node, and then create another packet for ZigBee based routing in sensor networks.

We also can directly query data from the interested sensor node as Figure 13 and 14. Internet users can directly query the data basing on *IPv6 address* or *Data Information*. However, generally the *Data Request* from Internet does not want to simply get data from one special sensor node, but needs the collaboration result of several sensor nodes. Therefore, we also can perform some attribute based query inside VIP Bridge as Figure 15 shows. Several sub-query commands can be created for different requested data based on the C*omplicated Data Request*.

After querying, packets that originally come from the sensor networks can also follow the following procedure to be sent back to IP based network user, as Figure 16 shows. VIP Bridge first bases on packet's ZigBee address to find out the created *T->S Packet*, then through the mapping between the created *T->S Packet* and the original *T->S Packet*, VIP Bridge can easily find out the original *T->S Packet*. By analyzing the original *T->S Packet*, VIP Bridge can get the *User IP*, and then create the new *S->T Packet*.

7 Discussion

7.1 How to Assign and Map IPv6 Addresses to Sensor Nodes?

After publishing paper [14, 15, 16], many researchers who do not have the background knowledge about the IPv6 stateless address auto-configuration [17, 18, 19] asked us the question that how these IPv6 addresses were assigned and mapped to sensor nodes. Basically, there are two different approaches: 1) *Explicit Assignment and Mapping*: VIP Bridge can use IPv6 stateless address auto-configuration function to assign a global unique IPv6 address to sensor node, this assigned IPv6 address do not include the sensor node's ID as a part of this IPv6 address, which is called postfix. The mapping between IPv6 address and sensor node ID is explicit. 2) *Implicit Assignment and Mapping*: In this approach the sensor node's ID is used as a postfix part of the IPv6 address, readers can refer [19] to know the detailed information. This kind of mapping between IPv6 address and sensor node ID is implicit. The sensor node ID can be extracted from corresponding assigned IPv6 address.

7.2 Key Features Differ VIP Bridge from Application-level Gateway

Many researchers also asked us the question that what the distinctive differences are between the traditional *Application-level Gateway* and our VIP Bridge, since our approach looks very similar to the former one. In this subsection, we present two key features to distinguish our VIP Bridge from *Application-level Gateway*.

Functionality: Figure 17 shows the logical location of our VIP Bridge. The A-CAMUS in the upper layer is another research project in our laboratory which is very similar to the *Application-level Gateway* approach. Readers can know more information about this project from [20]. We consider that gateways and bridges are two different ways to provide connectivity. Gateways provide a more full featured connectivity and allow a greater diversity of devices and applications to connect the ubiquitous sensor networks. However, ***bridges are much simpler than gateways*** and hence would be a lower cost to the user but serve a smaller application space. Here, our VIP Bridge has only one simple major function that is to connect heterogeneous sensor networks with IP based wire/wireless networks, and integrate these sensor networks into one virtual sensor networks.

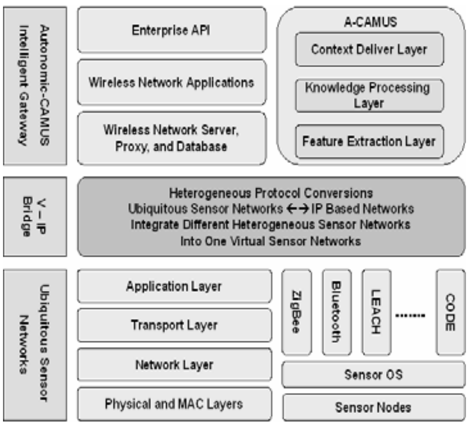


Fig. 17. Where VIP Bridge should be

Working model: In *Application-level Gateway*, the way to identify different sensor nodes is to use each sensor node's unique information, such as sensor node's ID or geographic location address. ***The query that issued by a certain end user basically targets to Application-level Gateway's IP address first, then after packet translation this packet will be forwarded to the targeted sensor node*** based on its ID or geographic location address. To accomplish this operation, the end user should be aware the Application-level Gateway's IP address and sensor node's unique information in advance, which is considered that the transparency for end user has already gone. However, in our VIP Bridge, the ***issued query will be routing directly based on the sensor node's unique IP address***. The only information that this user needs to know is the targeted IP address, but not including the sensor node's ID or geographic location address and VIP Bridge's IP address, which means it is totally transparent for end users. ***This changing will significantly influence the program-***

ming model of ubiquitous sensor networks, since application programmer can use only sensor node's IPv6 address and programmer' source IPv6 address to set up the TCP/IP socket.

7.3 Comparison with Related Researches

We think that a table based comparison with related researches is essentially necessary to prove that our solution can cover most of the benefit of related researches, as Figure 18 shows.

After the integration of sensor networks and IP based network, we can keep the consistency with the IP based working model by hiding the sensor ID or geographic location address. Because in the view of IP based network users, the sensor networks is IP based, they don't need to know which kind of routing protocol is used in sensor networks. In other words, sensor networks are transparent to Internet Users. However, for *sensor networks overlay TCP/IP,* users always have to deploy corresponding sensor networks routing protocol on Internet hosts, which means that users must know what kind of sensor networks they are.

	Application level gateways	Delay Tolerant Network	TCP/IP overlay sensor networks	Sensor networks overlay TCP/IP	VIP Bridge
Consistent with Internet working model	No	No	Yes	No	Yes
Transparent for Internet users	Yes	Yes	Yes	No	Yes
Freely choose routing protocol in sensor networks	Yes	Yes	No	Yes	Yes
Directly accessibility some special sensor node	No	No	Yes	Yes	Yes
Easy to integrate different sensor networks	No	Yes	No	Yes	Yes

Fig. 18. Comparison with related researches

Since we only deploy virtual IP addresses in bridge, rather than bring any modification to sensor networks protocols, sensor networks can still freely choose the optimized routing protocol which is *Node-Centric* or *Location-Centric* based. But the *TCP/IP overlay sensor networks* must modify the protocol stack of sensor networks.

Furthermore, Internet users can easily and directly access some special sensor nodes via *virtual IP addresses*. Since sensor networks can be *virtual-IP based*, it is very easy to integrate several locating in different place's sensor networks into one *virtual sensor networks*. ***Because we consider the integration of different sensor networks as a new research issue in the field of ubiquitous sensor networks, we are going to have more discussion about it in the following subsection.***

7.4 Integration of Different Sensor Networks

Sensor networks which are physically located in different locations may use totally different routing protocols for their specific applications, as Figure 19 shows.

Sometimes these sensor networks should be integrated into one virtual sensor networks over wired/wireless networks to provide comprehensive services for users.

In *Delay Tolerant Network,* because they deployed an additional *Bundle Layer* in both TCP/IP network and non-TCP/IP network protocol stacks, it is very easy to integrate different networks into one virtual network. However, it requests a lot of effort to modify existing routing protocols to deploy this new *Bundle Layer*.

In *sensor networks overlay TCP/IP*, if several sensor networks are only physically located in different locations but still use the same routing protocol, users can deploy this routing protocol to overlay TCP/IP networks, so that these sensor networks can be integrated into one virtual sensor networks. If these sensor networks are using different routing protocols, then this *sensor networks overlay TCP/IP* is not suitable to integrate them into one virtual sensor networks.

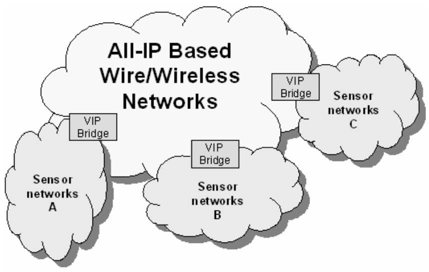


Fig. 19. Integration of several sensor networks

Compared with our VIP Bridge approach, either *Delay Tolerant Network* or *sensor networks overlay TCP/IP* needs to deploy or modify current existing protocol stacks. If these sensor networks have bridges which have *virtual IP address*es, then it is very easy to integrate them into one virtual sensor networks without any modification on existing protocols, because *virtual IP address* can hide all the heterogeneities of different sensor networks for upper layers.

7.5 Solution for previously mentioned VIP Bridge's limitation

In our previous papers [14, 15, 16], we mentioned that VIP Bridge has one limitation that the sensor networks which are not Node-Centric or Location-Centric based cannot be connected to IP based wire/wireless networks by using VIP Bridge. However, now we think that for those Data-Centric based sensor networks, we still can have some solution to let them use VIP Bridge, since the actual deployed every sensor node has its global unique MAC address, which can be used to match up with IPv6 address. We are going to go along with this direction deeply to see how exactly to make it be feasible as the future work.

8 Initial Implementation

In order to prove that our VIP Bridge can successfully support the integration func

CPU		Atmega128L
Internal Memory	SDRAM	4KByte
	FLASH	128KByte
RF		2.4GHz, ZigBee
Security		DSSS Supported
Low Power		AA Type Battery
Transmission Speed		250Kbps
Sensor Board		Included

Fig. 20. Hardware Platform: Nano-24 from Octacomm

tion, we built up our test bed to integrate three different ZigBee based sensor net works over IP based TCP/IP network into one virtual sensor network. We employ the

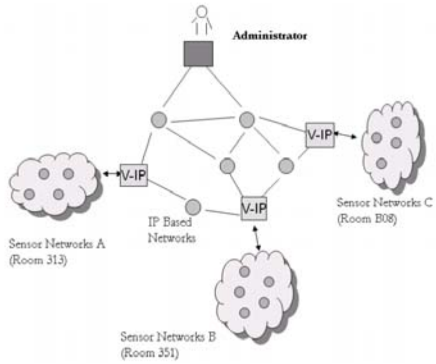


Fig. 21. Scenario: Office Monitoring

Nano-24 sensor board [21], developed by the Korean Octacomm company as our hardware platform. The hardware configuration is show as Figure 20. A collection of sensors are integrated on-board: 1) A light sensor for the detection of visible light; 2) A temperature sensor; 3) A humidity sensor. There is a separated ZigBee based communication board, which is designed for low-power applications. Two kinds of operating system can be deployed in this kind of sensor node: TinyOS [22] and Qplus [23]. TinyOS is a well-known open recourse operating system designed by UC Berkeley for wireless sensor networks. Qplus is an embedded Linux operating system developed by ETRI. Qplus consists of a reconfigurable embedded Linux kernel, system libraries, a graphic window system, and a target builder. ETRI and Red Hat cooperatively developed Qplus based on Red Hat's embedded software solutions and standards. In our implementation, we apply Qplus as our operating system.

The intended network scenario is a simple office monitoring. Sensor nodes are deployed in three separate rooms: Professor's Office (Room 313), Korean Students Research Lab (Room 351), Foreign Students Research Lab (Room B08). A lab administrator can use a website based interface to query the office environment status and decide whether to remotely turn on or turn off air-conditions for saving energy as Figure 21.

In order to create the view of virtual sensor networks, each VIP Bridge sends request to other VIP Bridges to ask for their mapping tables and finally every VIP Bridge will have all the mapping tables about other sensor networks as Figure 22.

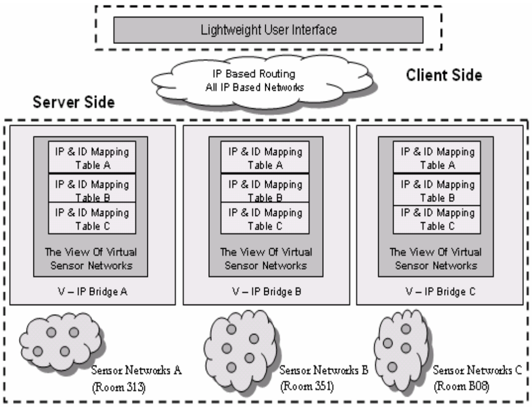


Fig. 22. The View of Virtual Sensor Networks

These mapping tables will be integrated into one, so that in users' side, through any VIP Bridge users are able to query sensor information over this whole virtual sensor networks by virtual IP addresses.

Readers are kindly invited to access our **on-going developing** VIP Bridge website to try the feasibility of our VIP Bridge through link: http://163.180.140.34.

9 Conclusion

Pervasive network which is considered as the next generation of current networks requests us to integrate all kind of heterogeneous networks into one global network. Sensor networks as a family member of wireless networks should be integrated with IP based wire/wireless network to provide meaningful services. In this paper we present the design and implementation works by using our VIP Bridge to connect sensor networks with IP based wire/wireless networks and integrate different sensor networks into one virtual sensor network. Here, we want to clearly point out that how to analyze one *Complicated Data Request* and create several *sub-Simple Data Requests* is another research issue, which is currently under investigation of another team in our group.

The significant contributions that made by our VIP Bridge can be summarized into following eight aspects: 1) successfully opened the door for leading ubiquitous sensor networks into the Next Generation Network paradigm; 2) first time successfully provided transparent accessibility for IP based wire/wireless network users to query information directly from any specific sensor node; 3) first time successfully changed the programming model for ubiquitous sensor networks' programmers; 4) first time successfully identified every sensor node in this world by using globally unique IPv6 address; 5) successfully connected sensor node ID based or geographic location address based heterogeneous sensor networks with IP based wire/wireless networks; 6) first time successfully integrated different heterogeneous sensor networks into one virtual sensor network; 7) successfully enabled the plug and play working model for any additional sensor node ID based or geographic location address based sensor networks to be integrated into virtual sensor network; 8) successfully provided comprehensive query methods for IP based wire/wireless network users.

References

1. Gary Legg, "Beyond 3G: The Changing Face of Cellular", http://www.techonline.com-/community/home/37977
2. Saha, D., Mukherjee, A. "Pervasive Computing: A Paradigm for the 21st Century", IEEE Computer, Volume: 36 Issue: 3, March 2003 Page(s): 25 – 31
3. N.Q. Hung, N.C. Ngoc, L.X. Hung, Shu Lei, and Sungyoung Lee, "A Survey on Middleware for Context-Awareness in Ubiquitous Computing Environments", Korean Information Processing Society Review ISSN 1226-9182 July 2003.
4. Z. Z. Marco, K. Bhaskar, "Integrating Future Large-scale Wireless Sensor Networks with the Internet", *USC Computer Science Technical Report CS 03-792*, 2003.
5. B. Sarikaya, "Serial forwarding approach to connecting TinyOS-based sensors to IPv6", http://tools.ietf.org/id/draft-sarikaya-6lowpan-forwarding-00.txt
6. Patrick Kinnery, "Gateways: Beyond the Sensor Network", ZigBee Alliance, http://www.zigbee.org/en/documents/SensorsExpo/7-Sensors-Expo-kinney.pdf
7. K. Fall, "A Delay-Tolerant Network Architecture for Challenged Internets". *In Proceedings of the SIGCOMM 2003 Conference*, 2003
8. A. Dunkels, J. Alonso, T. Voigt, H. Ritter, J. Schiller, "Connecting Wireless Sensornets with TCP/IP Networks", In *Proceedings of WWIC2004*, Germany, February 2004.
9. A. Dunkels, J. Alonso, T. Voigt, "Making TCP/IP Viable for Wireless Sensor Networks", In *Proceedings of the First European Workshop on Wireless Sensor Networks (EWSN'04), work-in-progress session*, Berlin, Germany, January 2004.
10. H. Dai, R. Han, "Unifying Micro Sensor Networks with the Internet via Overlay Networking", in *Proc. IEEE Emnets-I*, Nov. 2004.
11. C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks", Proc. ACM MobiCom'00, Boston, MA, 2000, pp. 56-67
12. L.X. Hung, and S.Y. Lee, "A Coordination-based Data Dissemination for Wireless Sensor Networks", ISSNIP 05, Melbourne, Australia December, 14-17, 2004.
13. ZigBee, URL: http://www.zigbee.org/en/index.asp
14. Shu Lei, Wang Jin, Xu Hui, Jinsung Cho, Sungyoung Lee, "Connecting Sensor Networks with TCP/IP Network", IWSN 2006, Harbin, China.
15. Shu Lei, Wu Xiaoling, Hui Xu, Yang Jie, Jinsung Cho, Sungyong Lee, "Connecting Heterogeneous Sensor Networks with IP Based Wire/Wireless Networks", SEUS 2006, Gyeongju, Korea, IEEE Computer Society

16. Shu Lei, Hui Xu, Wu Xiaoling, Zhang Lin, Jinsung Cho, Sungyoung Lee, “VIP Bridge: Integrating Several Sensor Networks into One Virtual Sensor Network”, ICISP 2006, France, Auguest 2006 URL: http://oslab.khu.ac.kr/vip or http://163.180.140.34
17. Montenegro, G. and N. Kushalnagar, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks", October 2005, URL: http://ietf.org/internet-drafts/draft-ietf-6lowpan-format-01.txt
18. Thomson, S., "IPv6 Stateless Address Autoconfiguration", draft-ietf-ipv6-rfc2462bis-07 (work in progress), December 2004.
19. T. Narten, R. Draves, “Privacy Extensions for Stateless Address Autoconfiguration in IPv6”, URL: http://www.ietf.org/rfc/rfc3041.txt
20. L. K. Saad, R. Maria, SungYoung Lee, YK Lee, "A Distributed Middleware Solution for Context Awareness in Ubiquitous Systems," RTCSA 2005, 17-19 August 05, HongKong
21. Octacomm Company, URL: http://www.octacomm.net
22. TinyOS, URL: http://www.tinyos.net/
23. Qplus, URL: http://qplus.or.kr/english/