P2P Mobile Sensor Networks

Srdjan Krco, David Cleary, Daryl Parker
R&D, LMI Ericsson, Ireland
E-mail: name.surname@ericsson.com

Abstract

Wireless sensor networks research is primarily focused on various internal sensor network issues like routing, self-organization, MAC layer design, data aggregation, etc. Interaction between sensor networks and especially such interaction over mobile networks has not been researched well. In this paper, concept of mobile peer-to-peer sensor networks, i.e. peer-to-peer sensor networks overlay on 3G mobile networks is presented. Each sensor network acts as one peer node and is represented by its gateway in a P2P network. Peers use mobile networks to communicate and collaborate on execution of specific tasks or to provide information to users that otherwise would not be available. Usability of JXTA as P2P framework in mobile environment has been investigated. Architecture of a sensor network gateway is also presented.

1. Introduction

Wireless sensor networks consist of a large number of tiny devices with sensing capabilities, able to perform simple data processing tasks and to communicate wirelessly with other similar devices. These networks are deployed in an ad-hoc manner, by scattering sensors across an area of interest. After deployment, sensor nodes initiate execution of self-organization algorithms to establish clusters, communication channels, hierarchy etc. Once established, it is expected that network would work months and more preferably years without human attention or need for replacement of nodes due to power resources depletion [1], [2].

Sensors communicate between themselves using short-range wireless communication links. Various choices are available: *Bluetooth* [3], *ZigBee* [4], simple RF communication, UWB (*Ultra Wide Band*) [5], etc. Each of the mentioned technologies has its advantages and disadvantages and depending on the application scenario and user requirements, one can be chosen over other. In case of small-scale networks that require standardized protocols, considerable data throughput, support for quality of service and accessibility from wide range of standard mobile communication devices, Bluetooth might be the choice. If a network has to support large number of sensors, very low energy consumption, medium data throughput and low initial connection setup delay, ZigBee is probably a better choice. Very high throughput and location service requirements will make UWB a very interesting candidate once it becomes widely available. Many of today's wireless sensor networks use simple RF with proprietary higher protocols layers built on top of it.

Users communicate with sensors via one or more dedicated nodes that act as so-called sink nodes. These nodes are responsible for injecting sensor queries into the network, gathering responses from sensors and forwarding them to users. As sensors communicate using short-range wireless technologies, in general case, it is not possible to establish direct communication links between all sensors in the network. Instead, multi-hop routes are used. Each sensor node executes routing algorithm and collaborate with other nodes to distribute user queries to relevant sensors and to forward sensors responses towards sink nodes. Sink node can communicate with users via a dedicate gateway node or directly if having dual sink/gateway node role (Figure 1).

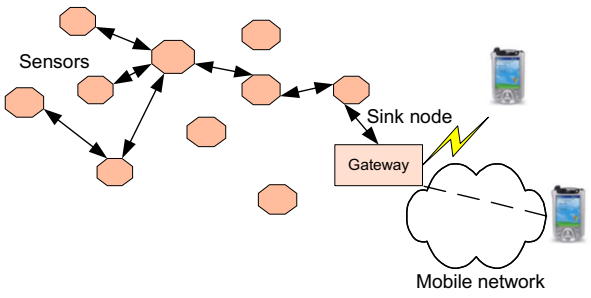

Figure 1 Wireless sensor networks architecture

Depending on the application scenario, short-range wireless communication links or wide-area network is used for communication between users and gateways. For example, in a closed environment like factory, technicians will mostly use Bluetooth or ZigBee to talk to gateways, while in environment monitoring applications users will

communicate with sensor networks over Internet or mobile networks.

Physical and functional constraints of wireless sensors limit their accuracy and sensing range. To overcome this handicap, several features of wireless sensor networks are used: wireless connectivity makes their deployment close to the observed phenomenon possible, small size, low cost and low energy consumption allow deployment of redundant number of sensors to monitor the same phenomenon and collaboration of sensors on execution of sensing tasks and aggregation of their observations increase accuracy and reliability of provided information.

This environment is obviously based on a different communication paradigm than conventional circuit switched communication or IP networking. In contrast to the conventional communication paradigm where communication links are most frequently established between two nodes, based on their unique network addresses, wireless sensor network users are not interested in establishing communication with a particular sensor. Instead, they look for certain information – which node or group of nodes will provide that information is of no relevance as long as the information is guaranteed to be accurate and reliable. Obviously, routing protocols designed for IP based ad-hoc environment are not suitable here [6], [7]. Instead, attribute based routing algorithms that disseminate user's data requests based on description of required data are used [8].

There are only a handful of, mainly military, real life applications that utilize large sensor networks today. Complex functionality in a constraint hardware and software environment presents a huge challenge to research community and is the main reason for slow resolution of many issues and slow deployment of large sensor networks. However, hardware and software platforms for development of sensor networks on a small scale are rapidly becoming available. In contrast to the large networks, these networks do not require complex self-organization, routing and other algorithms, which enables significantly easier deployment [9]. We envisage that as required technologies are becoming available on the market, individuals, companies and various organizations will rapidly start to deploy and use small-scale sensor networks. Each sensor network will consist of a handful of sensors and a gateway node connected to a mobile network. Providing remote health care, checking status of a vehicle, looking for a free car park space, checking level of air pollution around house, securing home, etc. are examples of possible applications. Some of the networks will provide information for private usage of their owners only. However, many of the networks will provide public services to all interested users. These users will interact with networks of interest on ad-hoc basis depending on their current needs.

Numerous networks will be used for same or similar purposes, i.e. will monitor the same phenomenon (traffic status at the current location of the network). Although independently and uncoordinatedly deployed, in many cases it would be beneficial from the provided service quality point of view, if networks could collaborate. Similarly to the collaboration principles of large sensor networks, these small networks can utilize common communication infrastructure provided by wide area mobile networks to communicate, collaborate and provide better coverage and more accurate and reliable information.

In this paper, architecture of wireless sensor network gateway is proposed and described. This gateway node hides internal sensor network implementation details and to users provides high-level description of services provided by the network. Potential solutions for collaboration of wireless sensor networks over mobile networks in a peer-to-peer fashion are also presented.

The rest of the paper is organized as follows. In the following section, concept of ubiquitous sensor networks is given. Architecture of a wireless sensor network gateway is given in Section 3. New concept of peer-to-peer networking of sensor networks over mobile networks is presented in Section 4. Section 5 describes JXTA P2P framework and its performance when deployed in mobile environment. Section 6 concludes the paper.

2. Ubiquitous sensor networks

Small-scale wireless sensor networks that do not require complex self-organization, multi-hop routing, data aggregation and similar algorithms are becoming readily available. Consisting of a relatively small number of sensors controlled by one central device, these networks will be carried on bodies, built into cars, buildings, attached to lampposts, etc. and will monitor health of their owners, traffic situation in an area, security of a house, air pollution in a street or quality of soil in a crop field. As enabling technologies are getting increasingly spread, various types of applications and usage scenarios will be developed and deployed.

Wireless communication interfaces will be used for interaction between users and networks. Depending on application scenario, short-range or wide area mobile communication technologies will be used. Persons wearing health care sensors will most certainly communicate with them using short-range wireless communication technology like Bluetooth or ZigBee. On the other hand, communication with home security system while away from home will be carried over a mobile network interface.

Practically, these networks will hang off mobile networks as independent sensor islands that move with

their hosts and provide services at different locations and at different times. Central devices that control each sensor network are the only "bridges" between users and sensors in a network, i.e. user can communicate with sensors only via these so called gateways nodes (Figure 2). A gateway node communicates with sensors over a short-range wireless interface on one side and with users over a short-range wireless interface or wide area mobile network interface on the other side and maps corresponding protocols. The further details about gateway node architecture and protocols are given in the next section.

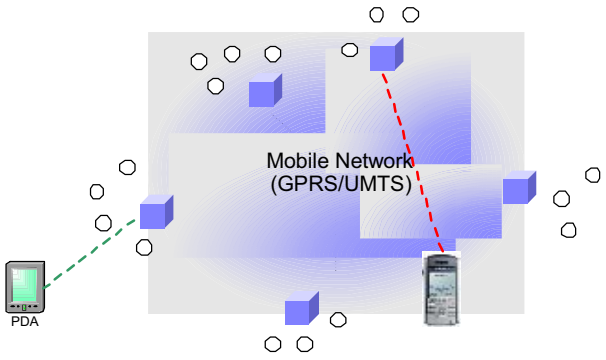

Figure 2 Small-scale wireless sensor networks

Many of the deployed networks will be providing very sensitive and personal information (monitoring personal health care status for example) and as such will be deployed for private usage only and accessible to a small, closed group of users. Since users and networks both will belong to one closed system, users will know how to establish communication with their networks, what type of information the network provides and will be able to use applications specifically designed to extract and present that information. As they will be operating in an open environment, prone to eavesdropping and possibly with a number of malicious users, appropriate security algorithms will be required to protect data integrity and privacy.

On the other hand, a number of network owners will be ready to share information gathered by their networks (for example traffic status at their current location) for mutual benefit of all involved parties or will deploy networks with the sole intention of providing services to interested users and charging for them. Some of these networks will be static (for example built into lampposts). Others will be mobile and will move with its owners, appear and disappear randomly, at different locations, in different contexts and at various time periods. In such environment where sensor networks come and go in an ad-hoc manner, deployed by numerous unrelated service operators, it will be impossible to establish a long lasting subscriber-operator relationship between sensor networks and their users. Users will not know about the existence of sensor networks in a certain area in advance nor will know what type of services discovered networks provide. Instead, depending on their current requirements and needs, users will have to use ad-hoc mechanisms to search for required services and available networks. Obviously, as variety of sensors and network types is enormous, both service discovery and communication protocols have to be very flexible and capable of supporting different types and formats of sensor data and services.

3. WSN gateway architecture

Small-scale wireless sensor networks consist of a relatively small number of sensor nodes and a gateway node. Gateway node is the central node in a sensor network responsible for establishment and maintenance of the network and is the only access point to its network. Unlike conventional wireless sensor networks where self-organization algorithms are executed by all sensor nodes, here that responsibility lies solely on the gateway nodes. It, in general, does not take part in sensing tasks, but is a dedicated control node, more powerful than regular sensor nodes, responsible for communication with users, as well as data collection and processing. Its significant data processing capability takes burden from sensor nodes and enables their simpler and cheaper architecture.

Depending on application scenario, users and gateways can communicate over a short-range or wide-area wireless interface. Sensors communicate with gateways over short-range wireless interface only. After establishment of communication links with sensors in a network, gateway nodes are then responsible for gathering information about the services these sensors provide and providing that information to potential sensor network users.

Apart from providing communication interfaces, the most important role a gateway has is the one of a sensor network representative. In other words, each gateway has the full knowledge of its network: what types of sensors are available, what services are offered, who can access sensor information, how to communicate with sensors, how to control them, collect data, etc. and is responsible for presenting these capabilities in a structured and logical manner to interested external parties.

Proposed system architecture utilizes attribute-based concept that does not use unique addressing schemes for establishment of communication routes. Instead, gateway provides description of available sensors and data they generate without providing addresses of sensor nodes or any other implementation details to users. Users can describe data they are interested in using provided description. It is responsibility of a gateway node to send requests for described data to relevant sensors in the network. All responses generated by these sensors are collected by gateway and forwarded to users as an integrated sensor network reply.

Apart from these basic functions, gateways also have to provide security solutions like verification of user credentials, granting appropriate rights to each user (querying only allowed, modification of parameters allowed, etc.), as well as procedures and algorithms required for service charging.

For the most of the sensor networks that are being carried by or wear by their operators, the most suitable platforms for implementation of gateway functionality are mobile phones and PDA devices. Their size, functionality, data processing capability and ubiquity make them an excellent platform for this purpose. In case of more static networks, small-scale PC or a similar microprocessor based devices can be used for implementation of gateways.

Proposed architecture of a wireless sensor network gateway is shown in Figure 3. It is a flexible architecture that supports various communication technologies, sensor types and user applications with no or minimal modifications.

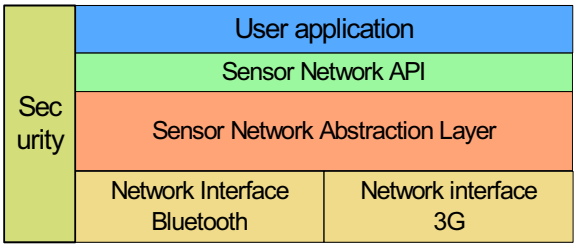

Figure 3 User control unit architecture

3.1. Wireless interface layer

The lowest layer in architecture is the communication layer. Usually, 2 different communication technologies will be supported by a gateway: one short-range and one wide range communication interface. Figure 3 shows Bluetooth and 3G as communication interfaces, but depending on the implementation various other interfaces like ZigBee, IEEE 802.16, 802.11, proprietary RF, GPRS etc. can be supported as well. The responsibilities of this layer are:

- To discover available wireless sensors in the surrounding environment when required;
- To establish reliable communication links with all sensors in the network;
- To execute routing protocol if such is required for forwarding information to and from sensors;
- To establish and maintain access to a mobile network using the most appropriate available service (SMS, MMS, audio/video streaming, TCP connection, etc.) for a given application.

3.2. Sensor network abstraction layer

Sensor network abstraction layer with its API represents the core functional entity of the proposed architecture. This is the layer that splits communication between sensors and users, encapsulates the internal organization of the sensor network and provides its abstract view to users. On this layer all sensors in a local sensor network are visible and accessible through their corresponding software objects. Each software object represents a sensor or a group of sensors and contains necessary methods for communication with them. API methods, based on user commands, use these objects to send appropriate messages to relevant sensors.

The responsibilities of this layer are as follows:

- Establish communication links with available sensors;
- Gather sensor profiles and compile that information into a form suitable for usage by remote users;
- Provide attribute-based addressing scheme for sensors in the network, i.e. allow users to describe data they need instead of requiring them to define from which specific sensors they need data;
- Based on user queries generate required protocol messages, forward them to relevant sensors, gather responses and forward sensor data to appropriate users;
- Map between sensor network API methods and communication protocol messages to enable control of sensors.

3.3. Sensor profile

A sensor profile is associated to each sensor in the system and it defines sensor's characteristics and its role in the system. Characteristics determine sensor type, accuracy, calibration date, manufacturer and other static sensor features that do not depend on the actual role the sensor has in the system. These parameters are usually set by sensor manufacturers. Parameters that define a sensor's role in a specific system, like "temperature sensor, first floor, room 22" or "motion sensor in the living room" are set by users.

Proposed format and structure of a sensor profile are presented below. Sensor profile is a flexible XML structure adaptable to a specific user and application requirements. It consists of two parts: *attributeProfile* and *dataProfile*.

attributeProfile describes basic sensor characteristics like sensor type, location, manufacturer, accuracy, measurement range, calibration date, A/D resolution, etc. The second part of the profile, *dataProfile*, describes format of data the sensor is generating, i.e. is it an integer or a float number, a single number or sequence of numbers, how many bytes are used to represent one

sample, etc. An example of a sensor profile describing a temperature sensor located on the second floor, room 22 follows.

```
<?xml version="1.0" encoding="UTF-8"?>
<sensor>
  <attributeProfile>
    <attribute name = "type">
      temperature
    </attribute>
    <attribute name="location">
      <attribute name="building">
            Radio house
      </attribute>
      <attribute name="floor">
            2
      </attribute>
      <attribute name="room">
            22
      </attribute>
    </attribute>
  </attributeProfile>
  <dataProfile>
      <dataAttribute>
        <Adresolution>
            12
        </Adresolution>
        <samplingRate>
            200
        </samplingRate>
        <sampleSize>
            4
        </sampleSize>
      </dataAttribute>
    </dataProfile>
</sensor>
```

When a gateway discovers a new wireless sensor, it collects profiles of all sensors that belong to the discovered wireless sensor first. Based on the information provided in the profiles, gateway then creates and configures objects and parameters required for providing access to sensors and information about available sensor data.

4. Mobile P2P sensor networks

The deployment of mobile sensor networks for mutual benefit of all involved parties or will deploy networks with the sole intention of providing publicly available services and then charging for them. In these scenarios interaction will be in the form of a P2P communication model. These newly formed networks will fall into to categories, static and dynamic in nature. Static configured (for example built into lampposts), while others will be mobile and will move with its owners, pop out and then disappear randomly, while providing information at different locations, in different contexts and at various time periods. In such environment where sensor networks come and go in an ad-hoc manner, it will be impossible to establish long lasting relationship between sensor networks and their users.

Users interaction with external sensors networks will occur in a spontaneous fashion. This session concept will depend on their current requirements and needs, realization of this dynamic service formation with be facilitated by ad-hoc mechanisms to search for required services and available networks. Obviously, as variety of sensor and network types is enormous, both service discovery and communication protocols have to be very flexible and capable of supporting different types of information and services.

Further on, individual sensor networks will be deployed by their respective owners independently of each other, which will undoubtedly lead to the existence of numerous networks that provide the same or similar services at the same or different locations. Leveraging on the fact that all of them are connected to mobile networks, individual sensor networks and their users can benefit from sharing available information and collaborating on providing better informed service based on combined input from all available networks. For example, sensor networks built into cars can communicate information on traffic status at their current location thus covering a much larger area. Air pollution sensors built into lampposts and buildings can collaborate to provide air pollution footprint covering a much larger area. Personal health care sensor networks carried on patient's bodies can interact to search for a medical advice or to establish a support forum for overcoming consequences of a specific illness.

This concept basically presents the peer-to-peer networking concept that is widely in use today in fixed communication networks [15, 16], but mapped to mobile environment. Each sensor network presents a peer node capable of working and providing information independently of other peers, but also of communicating with other nodes and sharing available information with them. Collaboration of completely uncoordinated and nomadic networks on execution of a common task in a mobile environment is obviously not easy to implement. Different types of information and services, various data formats and application requirements, connectivity of and ability to discover sensor networks connected to different mobile networks are some of the most interesting issues.
This concept presents a contrast to the conventional wireless sensor networks. Conventional networks are deployed by a single entity and are large standalone networks of complex nodes that use complex protocols

for internal network communication and collaboration. Peer-to-peer networking concept described above deals with a number of smaller networks deployed by various entities and with rather simple architecture that leverage on the existence of the communication infrastructure provided by mobile networks to communicate and collaborate.

4.1. Implementation issues

As stated in the previous sections, individual sensor networks will be deployed randomly, based on network owners plans, intentions or needs. Availability of a certain sensor network thus changes unpredictably and each time users want to use a certain service they have to search for an available network that provides required service.

Two scenarios should be observed:

- Users and sensor network gateways communicate over a short-range wireless communication technology;
- Users and sensor network gateways communicate over wide area mobile networks.

The former case occurs when users are looking for specific services available at their current location and in their current context. The first step in this process is detection of available networks in the vicinity. It is achievable using appropriate functionality of the underlying communication technology, like Bluetooth's Inquiry procedure [3]. After detecting a network, a service discovery protocol is invoked to check if detected networks actually provide services user is interested in.

In the latter case, users are looking for specific services provided by sensor networks at their current or, more probably, some arbitrary remote location. Detection of a sensor network or, in general case, a mobile network subscriber at a specific location is not possible without help from mobile infrastructure and nowadays is provided on limited basis and as a separate service. Today, mobile subscribers are described with their subscriber numbers only and information about services that users might be offering is not readily available. Hence, even if mobile network would provide connection parameters of subscribers at a given location, it would not be possible to determine which subscriber actually offers sensor network services. The least that mobile network could do to facilitate a simple service discovery is to flag those subscriber numbers that actually represent sensor networks based on information provided by subscribers at the subscription time. This would enable users to contact each of the provided subscribers and to query them separately for required services. Obviously, such interaction would be very inefficient, would cause a lot of unnecessary traffic, would not guarantee discovery of a service even if the service really existed and would not allow dynamic changes of the service portfolio.

A much better approach would be provision of a service discovery service by mobile operators. This service would have to provide support for registering available services and querying listed services. Based on the query results, user would get subscriber numbers of all sensor networks that can provide requested information. It would be then up to each user to contact provided networks and to combine their inputs into the final answer.

Our approach is based on peer-to-peer networking concept described in the previous section. Each sensor network represents a peer node in a P2P network overlay on a mobile network. Using P2P protocols and procedures sensor networks can publish available services and collaborate on execution of user-defined tasks, while users can query all peer nodes easily and search for available services. In the next section we will highlight the JXTA P2P middleware and adaptation we used to create an effective abstract network overlay.

5. JXTA

At the center of the JXTA design is the suite of protocols that comprise the core of the JXTA platform [12]. Figure 4 illustrates how the Peer-2-Peer paradigm is realized as an overlay abstraction on top of physical communication networks. JXTA protocols are based on an asynchronous query/response model. The message structure is a self-describing metadata format defined in XML [10]. Unlike most query/response models [11] the JXTA protocols do not have a one to one relationship between the queries and responses. The JXTA protocols are designed to be quite flexible, a query may receive zero, one or many responses depending on the nature of the query and the structure of the peer network.

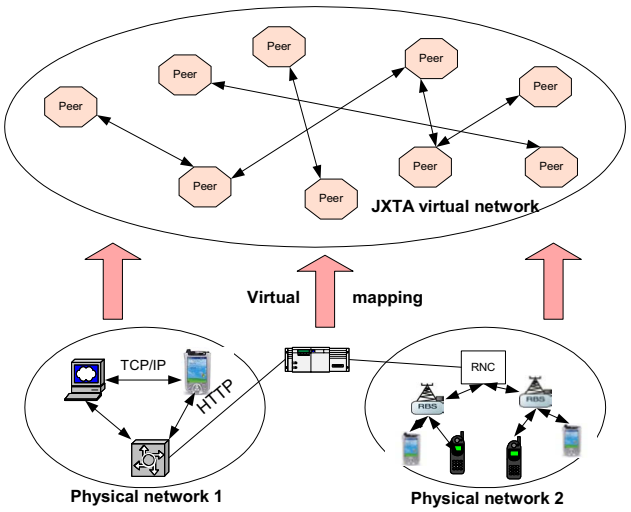

Figure 4 JXTA virtual network

In Figure 5 we present the structure of the JXTA service stack. The *EndpointService* provides the transport layer to send and receive messages to both the *PipeService* and the *RendezVousService*. The *RendezVousService* provides the *ResolverService* with the necessary mechanisms to propagate messages within the peer group. The *DiscoveryService*, *PeerInfoService*, and *PipeService* in turn rely on the *ResolverService* to envelope their protocol messages within the generic Query/Response structure it provides. As evidenced by the above diagram there is a neat layering of the services, which is broken only by the *PipeService*. It is slightly unusual in that it interacts directly with both the *ResolverService* and the *EndpointService*.

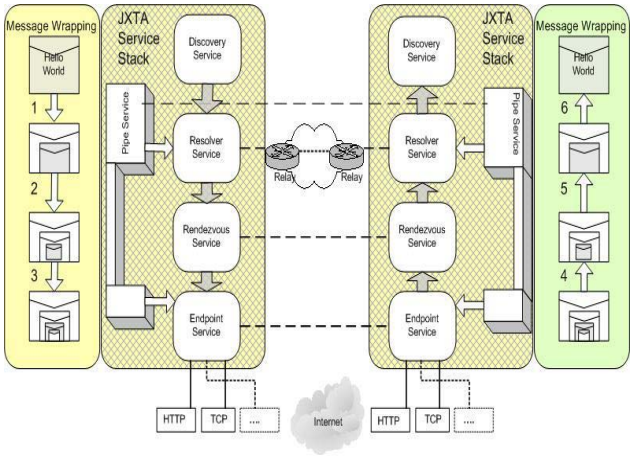

Figure 5 JXTA Stack and message encapsulation

To understand the message layering that occurs at each level in the stack the following example is presented. A generic Advertisement or XML document is constructed that contains the "Hello World" contents. The DiscoveryService will wrap this inside a DiscoveryResponse message, which is then passed to the ResolverService (1). The ResolverService in turn envelops the DiscoveryResponse in a generic ResolverResponse to be passed to the RendezVousService (2). The RendezVous service proceeds to wrap the ResolverResponse into a PropagateMessage for propagation in the network (3). Before transport the EndpointService will further append an EndpointHeaderSrcPeer element defining the peer from which this message is originating. On receipt of the message at another peer, there is a reverse process of unwrapping the various protocol messages (4 - 6), to reveal the original contents. The interaction details at each layer are described in detail in the following sections.

5.1. JXTA Services Layer

As well as the layered architecture and specific protocols of JXTA it is helpful to familiarize oneself with some of the key JXTA constructs in order to understand our proposed mobile P2P architecture. We have grouped these constructs into three key areas, communication, data abstraction and services.

Communications

A peer is a logical component uniquely identified by a peerID. Peers are not tied to a specific machine or user, and can operate independently and asynchronously from all other peers. Peers communicate via asynchronous, uni-directional message transfer mechanisms know as pipes. Pipe are dynamically bound to peer endpoints which correspond to available peer network interfaces, i.e. TCP port and IP address, that can be used to send and receive messages. Messages may pass through several intermediary peers before arriving at their final destination. A group mechanism is also supported where only particular services provide the required control to the resources of the group. The JXTA Protocols specifies how peers may publish, discover, join and monitor peer groups.

Data Abstraction

All JXTA resources such as peers, peer groups, pipes and services, are represented by *advertisements*. These are language neutral metadata structures represented as XML documents. Advertisements are generally exchanged via the ResolverService on behalf of higher layer services. Alternatively, data may be exchanged via pipes using Messages. Messages are the basic units of data exchange over pipes and consist of ordered sequence of named and arbitrarily typed values called message elements i.e. a set of name/value pairs.

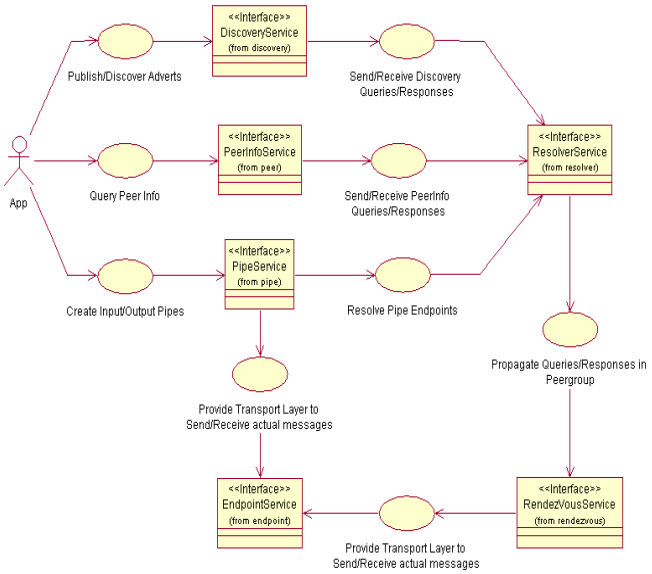

Figure 6 System Level Use Case

Services

Peers cooperate and communicate to publish, discover, and invoke *network services*. There are two levels of network services defined by JXTA. Peer Services, are only accessible on the peer publishing the service i.e. the service instance is scoped to that peer.

The second set of services is referred to as PeerGroup Services. These services have replicated instances on all peers in the group and are collectively identified by a unique id. The platform layer, also known as the JXTA core, defines the essential elements of the protocol suite. Access to these protocols is made available through a set of core PeerGroup services as depicted in the system use case view of Figure 6.

5.2. JXTA experiences

In order to evaluate the use of JXTA within a mobile network environment, we have implemented a test bed consisting of the following elements:

- Wireless sensor nodes – small, portable, microcontroller based devices with a Bluetooth interface for communication with sensor network gateways
- Sensor network gateway – Linux based laptops equipped with Bluetooth and GPRS interfaces. Each gateway also functioned as a JXTA peer using a custom Endpoint binding running over GPRS
- Peer Rendezvous node – a Rendezvous node was added to the private GPRS network to provide interconnects between mobile peers, due to the lack of multicast support in GPRS

Initially, it was planned to use PDA devices as sensor network gateway/peer nodes, but due to the immaturity of the J2ME version of JXTA at the time this functionality was instead implemented on a laptop. Local connectivity of the sensors was achieved via Bluetooth, using the JSR 82 compliant Atinav stack. Wide area connectivity was supplied via GPRS modems. This led to a number of interesting technical challenges. JXTA's discovery mechanisms rely on the use of IP multicast, which is not supported by GPRS. Thus a special Rendezvous was configured within the GPRS private network, to which all gateway peers were registered. The address of the Rendezvous node was configured dynamically via an SMS message on registration with the network. The setting up of a PDP context for the GPRS connection was implemented in a similar manner. To hide the complexity of this configuration and to ensure correct operation, a custom GPRS Endpoint binding was implemented.

The goal of our experiments was to determine if JXTA can be used in mobile environment and to get the initial feeling of its performance in this environment. During our tests there was no background traffic in the network. Typical experiments consisted of four peers with 2-3 sensors each, exchanging data. Standard GPRS throughput achievable in downlink direction in commercial networks is between 33kbps to 56kbps, depending on the number of timeslots available to user equipment. We observed that due to the verbose nature of the XML based protocols used in JXTA, coupled with the poor implementation of the routing mechanisms, the actual throughput available to user applications was reduced to just a few kilobits per second. Introducing a compressed version of the JXTA messages reduced this overhead quite drastically but the sheer number of messages sent as a result of the routing implementation showed little overall improvements.

The observations regarding performance of JXTA mentioned in this paper are based on the JXTA 1.1 Reference Implementation and as such many have been addressed in later releases of the platform. Recent work based on the JXTA 2.1 reference implementation has shown the performance of the JXTA messaging layer to be adequate for Near Real time applications [13]. This is as a result of major improvements in the platform, including a binary messaging format, improved rendezvous implementation based upon a distributed hash table [14].

6. Conclusion

Technologies for implementation and deployment of small-scale wireless sensor networks with mobile interface (3G) are becoming available. Once deployment of such networks start, we will increasingly become surrounded with various, static and mobile, sensor networks deployed by various individuals and organizations. We will interact with such networks opportunistically in order to satisfy our needs at a given location and at a given moment.

Architecture and protocols proposed in this paper support such ad-hoc way of communication and are capable of supporting arbitrary types of communication technologies and sensor services. Attribute-based approach and XML based description of sensor services enable users to get comprehensive and structured information about available services and to request required data in a similar manner.

As networks will work in environment with potentially large number of malicious users, additional work is required to develop appropriate data security and integrity solutions.

Proposed architecture also supports collaboration of individual sensor networks over wide area mobile networks. Using JXTA overlay on a GPRS network to

connect sensor networks in a peer-to-peer manner, it is possible to establish groups of networks based on their common interests. Such collaboration secures better service area coverage and provides means for sharing information between the networks. Performance of the initial JXTA implementation was not good in mobile environment, but improvements made in later implementations substantially improve its performance and to a great extent make JXTA suitable for usage in mobile environment.

Acknowledgment

Authors would like to thank all Skylark project members for their valuable contribution to this paper.

7. References

[1] Chee-Yee Chong, S. P. Kumar, "Sensor Networks: Evolution, Opportunities, and Challenges", Proceedings of the IEEE, Vol. 91, No. 8, August 2003

[2] G. Asada, et al., "Wireless Integrated Network Sensors: Low Power Systems on a chip," Proc. of the European Solid State Circuits Conference, 1998.

[3] Bluetooth v1.1 specification, Bluetooth SIG, www.bluetooth.org

[4] IEEE 802.15.4 Specification

[5] D. Porcino, W. Hirt, "Ultra-Wideband Radio Technology: Potential and Challenges Ahead", IEEE Communications Magazine, July 2003

[6] C. Perkins, "Ad hoc networking", Addison-Wesley, December 2000

[7] IETF MANET working group, http://www.ietf.org/html.charters/manet-charter.html

[8] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, F. Silva, "Directed Diffusion for Wireless Sensor Networking", IEEE/ACM Transactions on Networking, Vol. 11, No. 1, February 2003

[9] S. Krco, "Bluetooth Based Wireless Sensor Networks – Implementation Issues and Solutions", Invited paper, Proc. of the Telfor02, Belgrade, Serbia&Montenegro, Nov. 2002

[10] F. Yergeau, J. Cowan, T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler, Extensible Markup Language (XML) 1.1, XML 1.1, W3C Recommendation, 4th February 2004

[11] Gudgin, Hadley, Mendelsohn, Moreau, Frystyk, Nielsen SOAP Version 1.2 Part 1: Messaging Framework W3C Candidate Recommendation 19 December 2002 http://www.w3.org/TR/2002/CR-soap12-part1-20021219

[12] JXTA v2.0 Protocols Specification, http://spec.jxta.org/nonav/v1.0/docbook/JXTAProtocols.html

[13] D. Parker, S. Collins, D. Cleary, "Building Near Real-Time Peer-to-Peer Applications with JXTA", 4th International Workshop on Global and P2P computing, April 2004.

[14] B. Traversat, M. Abdelaziz, E. Pouyoul, **"**A Loosely-Consistent DHT Rendezvous Walker", Sun Microsystems, Inc., March2003

[15] http://gnutella.wego.com

[16] http://www.napster.com