jxSensor: a sensor network integration layer for JXTA

Marc Domingo-Prieto, Joan Arnedo-Moreno
Internet Interdisciplinary Institute (IN3)
Universitat Oberta de Catalunya
C\Roc Boronat, 117 , 7 flour 08018 Barcelona, Spain
mdomingopr, jarnedo@uoc.edu

Xavi Vilajosana-Guillén
Computer Science, Multimedia and Telecommunication
Department, Universitat Oberta de Catalunya
Rambla del Poblenou 156, 08018 Barcelona, Spain
xvilajosana@uoc.edu

***Abstract*—Nowadays, Wireless Sensor Networks (WSN) are already a very important data source to obtain data about the environment. Thus, they are key to the creation of Cyber-Physical Systems (CPS). Given the popularity of P2P middlewares as a means to efficiently process information and distribute services, being able to integrate them to WSN's is an interesting proposal. JXTA is a widely used P2P middleware that allows peers to easily exchange information, heavily relying on its main architectural highlight, the capability to organize peers with common interests into peer groups. However, right now, approaches to integrate WSNs to a JXTA network seldom take advantage of peer groups. For this reason, in this paper we present *jxSensor*, an integration layer for sensor motes which facilitates the deployment of CPS's under this architecture. This integration has been done taking into account JXTA's idiosyncrasies and proposing novel ideas, such as the *Virtual Peer*, a group of sensors that acts as a single entity within the peer group context.**
[1]

Keywords: peer-to-peer, WSN, JXTA, Java, sensor mote, Cyber-Physical System.

I. Introduction

In the current networked society, Wireless Sensor Networks (WSN's) have become very important, since they can acquire and send really useful and valuable information with a lesser cost of installation and maintenance, compared with their wired counterparts. The benefits of WSN's are not only in economical terms but also in functional ones, allowing an easy and fast deployment, allowing the mobility of nodes and dynamic network topologies, which has a big impact in temporal installations. WSN's have been traditionally related to industries such as military, oil and gas, but as their popularity has increased, they have become ubiquitous in many other fields and are already part of our daily life [1].

WSN's are able to provide detailed information about the environment, and therefore, are the key to the deployment of Cyber-Physical Systems (CPS), being their main means for data input. However, it is convenient to have some middleware that integrates access and configuration of sensor motes, overcoming the two common limitations in WSN's: that a great amount of information must be distributed and processed by constrained resources and the fact that, even though such information can be easily gathered locally, global access is often necessary.

Peer-to-peer (P2P) networks are an efficient method to distribute information in a self organized manner. These networks also provide incredible benefits in terms of the connectivity of its devices, such as scalability and reliability. For these reasons, we consider that the P2P systems might be a very good fit for developing CPS's in some environments. Many different P2P middlewares have become quite popular, helping the deployment of such networks, but between the most well-known ones with a long history, JXTA can be found [2]. JXTA is a set of open protocols specifications initiated by Sun Microsystems in 2001 that has been the system of choice for many P2P based applications, such as clipboard and file sharing between different computers [3] and a distributed e-learning system [4]. These applications can take advantage of the integration of WSN. Still under development, the latest revision at this date, version 2.7, became available in May 2011 and incorporates several long awaited functionalities, the most relevant improvements being in security and the simplification of local deployments for the testing stage.

This paper presents *jxSensor*, an abstraction layer that allows JXTA peers to interact with WSN's, facilitating the deployment of CPS's under this architecture. Even though extensive work already exists on the use of JXTA as a P2P substrate to distribute sensor information, our proposal uses a novel approach that sets it apart from others: instead of considering sensor motes as resources to be shared inside the JXTA network, they are considered full fledged peers. This small twist allows the main contributions of this paper: it takes into consideration actual two-way interaction, allowing active configuration of sensor motes, and the WSN gateways' operation remains completely in the background. From the other network participants' standpoint, sensor motes appear as a normal JXTA peer, allowing the use of some very useful JXTA capabilities which are not considered in current proposals, such as group membership across multiple gateways.

Following, we introduce the structure of this paper. First of all, Section II performs a literature review with all the relevant work on P2P-WSN integration which relies on JXTA as the underlying P2P middleware. The description of our proposed architecture and its underlying modules is included in Section

[1]This work was partly funded by the Spanish Government through projects TSI2007-65406-C03-03 "E-AEGIS", TIN2011-27076-C03-02 "CO-PRIVACY" and CONSOLIDER INGENIO 2010 CSD2007-0004 "ARES".

978-0-7695-4779-4/12 $26.00 © 2012 IEEE
DOI 10.1109/NBiS.2012.125

III. Section IV specifies the protocol for both the P2P and the WSN sensor side of the integration layer. Finally, Section V concludes this paper with a brief discussion about the paper's main contributions and future work.

II. Related Work

In this section, we provide a brief literature review on current proposals which rely on a P2P middleware to distribute WSN data. Even though proposals using different middleware exist [5], [6], we will only focus on those which specifically use JXTA, so it is possible to analyze how JXTA's particular capabilities are integrated into each approach. A full review of such capabilities and JXTA's architectural design can be found in [2].

One of the most thorough works may be found in [7]. In this paper, the authors propose a quite complex general purpose architecture, even though VANETs are the main scenario in mind, for sensor mote data acquisition over a JXTA middleware overlay operating on 2.5G/3G mobile devices. This is achieved by heavily modifying the standard JXTA distribution, thus creating an alternate version. It must be told that the paper gives much more emphasis to the mobile device integration aspect of the system. Nevertheless, from the WSN integration standpoint, the most interesting contribution is the design of the XMLSens protocol, which allows sensor motes to announce their characteristics to the WSN gateway using XML structures, in a manner similar to JXTA's lower layer protocols.

In [8], JXTA is again the middleware of choice to distribute sensor mote data, in this case in a healthcare services scenario. Sensor mote data from a Body Area Network are aggregated using a PDA, which then relies on a JXTA relay peer, acting as a proxy, to access the P2P network. However, not much emphasis is done on the WSN part of the system. What makes this proposal specially relevant in the context of a literature review is the fact that it is the only approach where JXTA peer groups are fully considered as a natural method to segment the P2P network, from both an efficiency and security standpoint.

Sharesense, a P2P environment based on JXTA for monitoring multiple WSN's is presented in [9]. At its core concept, its approach relies on integrating JXTA to jWebDust, an external Java environment, previously proposed by the same authors, that allows developing and managing WSN based applications . This environment, being Java based, is easy to integrate with the Java implementation of JXTA, and from the system's architecture, seems to be the one doing the heavyweight work as far as the sensor mote's data management and access is concerned. Thus, the system can only integrate sensor mote networks based on this particular environment. Additionally, a Google Earth-based interface is included in the demo application, allowing precise location of each sensor mote.

Nevertheless, apart from the specifics of each proposal, some common features are shared by all of them to some degree. Sensor motes are always considered as resources to be shared, and two of JXTA's core services are mostly used to access their data. On one hand, JXTA's Discovery Service is used to publish and locate available sensor motes or WSN's, relying on advertisements, XML metadata documents, to describe the sensor mote's characteristics. On that regard, each proposal provides its own custom-made advertisement structure. On the other hand, the WSN gateway acts as a single peer in the P2P substrate, relying on JXTA's Pipe Service to receive messages from other peers. This is a sound and straightforward way to integrate the JXTA middleware to a set of WSN's from a design standpoint.

However, it is important to note that, except in a single case, current proposals completely forget, or use in a very rudimentary way, another of JXTA's main architectural features, and the one which actually sets it apart from other P2P middlewares: peer groups. We deem taking JXTA peer groups into consideration important since they allow peers with similar capabilities to create a context for peer operation, segmenting the P2P network and facilitating advertisement publication and retrieval.

III. jxSensor architecture overview

The general structure of the jxSensor network at a logical and physical level is shown in Figure 1. This figure serves as an overview of how jxSensor is integrated within the context of the JXTA and WSN layers. The whole P2P-sensor integration layer is executed in a sensor gateway, a specialized hardware that acts as a bridge between the local WSN environment and external networks, such as the Internet. This aspect is imposed by the design of the WSN's. One example of such devices is the alix3d3 [10]. On that regard, on one hand, at the JXTA side of the architecture, applications which desire to obtain sensor data, usually the core behind a CPS, are executed in peers at the JXTA Applications layer. On the other hand, at the WSN side, the sensor motes execute ordinary Sensor Apps, capturing and sending data, and receiving and executing actions. These actions include changing the configuration of the application or requesting data from a sensor. In both kind of applications, their behavior is up to the developers, and does not have to be specific to jxSensor.

The hardware gateway executes the main P2P-WSN integration module, **WSNGateway**, at the JXTA's community services layer. As a result, it is easily pluggable to the standard distribution of JXTA 2.7 without the need to modify the source code, which needs the creation of a custom made JXTA version. Furthermore, it is not necessary to install a specific client component in other peers before they can actually communicate with the sensor motes. Any JXTA Application may interact with the WSNGateway module using the standard JXTA primitives for peer location and message exchange. This module is composed by the following components:

Virtual Peer: Acts as a peer entity within the JXTA network on behalf of one or a group of sensor motes, storing and managing all the information an actual JXTA peer has (for example, a unique JXTA ID, advertisements, etc.). A single WSNGateway may support several Virtual Peers, and thus,

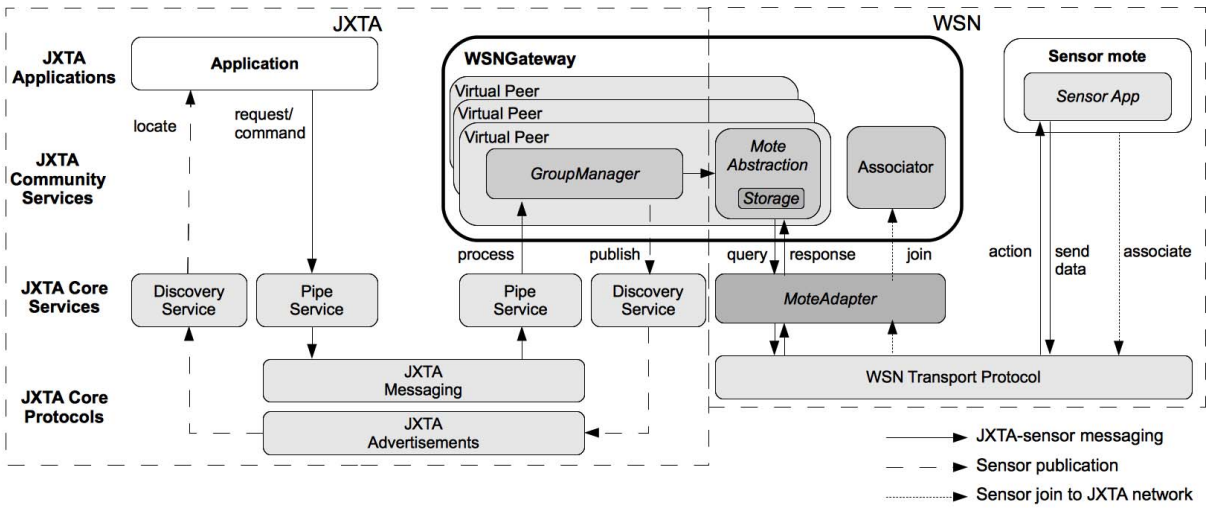

Fig. 1. Overview of jxSensor's modular architecture

several instances of this component may be executing concurrently at any given time, each one representing a separate sensor mote proxy. A set of motes may be managed using a single Virtual Peer, which groups all of them and may be interacted with a single entity. Thus, configuration commands are applied to all sensor motes at a time, and the answer of a query can be resolved by any sensor mote in the group, allowing for example to distribute the energy consumption.

GroupManager: Manages all data regarding a Virtual Peer's membership to a particular JXTA peer group. A Virtual Peer may belong to several JXTA groups, and therefore may deploy several GroupManagers.

Mote Abstraction: Processes queries received by a Virtual Peer from the JXTA network, requesting actions on sensor motes if necessary. If a query requests current data from a sensor mote, it is forwarded to the sensor mote, its response is cached to an internal storage and then sent to the requester. In the case of a query for historical data , the data is directly retrieved from the cache, without the need to query the actual sensor. Access control to the WSN, when required, is also enforced at this component.

Associator: Allows sensor motes to associate with the gateway, requesting the creation of a Virtual Peer that proxies them within the context of the JXTA network.

At a lower level layer, the **MoteAdapter** service is a pluggable component that acts as the abstraction layer that translates queries to the actual protocols accepted by Sensor Apps deployed at the WSN. This is one of the most important modules, decoupling the primitives processed at the MoteAbstraction component and the sensor motes. The implementation of the MoteAdapter service is specific for every WSN protocol and application used, providing high flexibility, since all the heavy work falls on the gateway, instead of each sensor mote.

A clearer idea of each component's functionality may be expressed with a description of a sensor mote's operation under the jxSensor architecture. Whenever a sensor mote (or a group of them) desires to be accessible from the JXTA network, first of all, its Sensor App must send its sensing capabilities and the JXTA groups it wants to be member of to the WSNGateway's Associator component. This data is transmitted using the WSN Transport Protocol and translated by the MoteAdapter service, which will forward the petition to the Associator component. As a result, a new Virtual Peer is created, which immediately joins the specified peer groups. A GroupManager component is deployed within the Virtual Peer for each joined group, which will manage JXTA data exchanges within the context of that group. The sensor mote is then considered associated to the WSNGateway, which will proxy all data exchanges with the JXTA network via the new Virtual Peer.

At this point, the sensor mote is available on the JXTA network and can be discovered and accessed using JXTA primitives as any other peer would be. Once discovered, any other peer can send requests for data retrieval or configuration commands using JXTA's standard messaging capabilities, which are managed by the WSNGateway and processed by the corresponding Virtual Peer. Whenever a request implies communications with a physical sensor mote device, it is sent to the mote by using the MoteAdapter service. Responses are then sent back to the JXTA network also via the corresponding Virtual Peer component.

Even though our proposal still follows some of the most basic JXTA integration architectural approaches used in the related literature, as described at the end of Section II, there are still some important differences. In our case, sensor motes become part of the network as peer themselves, instead of simply resources shared by the gateway. The main benefits of this decision are twofold.

First of all, if a sensor mote is considered a resource, in order to access its data the following steps must be followed:

1) Locate the sensor mote with the chosen sensing capabilities using the Discovery Service.
2) Locate the gateway, also using the Discovery Service.
3) Request the sensor mote's data using the Pipe Service.

However, under our scenario, step 2 may be completely omitted. Once the sensor mote of choice has been located, it is possible to directly send a request. Obviously, the gateway is still acting as a translator in the background, since sensor motes, due to their underlaying protocols and limited nature, are not actual JXTA peers. But it is not necessary to explicitly look it up, sending additional JXTA discovery requests.

Secondly, considering sensor motes as JXTA peers allows the full use of JXTA's Peer Group capabilities. Every sensor mote may become a member of any peer group, and most importantly, each sensor mote's membership is not restricted by its geographical location, which is the case in the only existing proposal where peer groups are even considered. Sensor motes connected to different gateways can share the same group, providing a larger flexibility in JXTA network segmentation. For example, seismographic sensor motes spread across disparate locations may be configured into a single peer group, so applications interested in only such data may locate them in a much more simpler and efficient manner. In addition, our approach allows sensor motes to become members of several groups, not just a single one.

IV. JXSENSOR PROTOCOL SPECIFICATION

This section presents the specification of the protocols used by jxSensor at the JXTA and WSN layers. On one hand, the former case details how individual sensor motes or groups of them can be located within the context of a JXTA peer group, as well as message syntax to send and retrieve information to/from them. On the other hand, the latter scenario details the message exchange between the Mote Abstraction component and the MoteAdapter service, not sensor motes. This is because, as mentioned before, the MoteApdapter service provides the abstraction layer which translates queries/responses to the format used by the non-standard Sensor Apps deployed at the WSN.

A. P2P layer protocols

jxSensor relies on three basic operations at the JXTA layer, all executed by JXTA peers: discovering existing sensor motes in a peer group, sending data requests to them and applying new configuration to sensor motes.

1) Sensor mote discovery: The JXTA presence mechanism within a peer group are Peer Advertisements, metadata XML documents which offer a basic description of a peer (JXTA ID, name and description), the means to reach them (routing information) and a list of service parameters, which usually include Pipe Advertisements, the necessary information to establish connections with each particular service being executed at that peer. In jxSensor, a Virtual Peer's GroupManager component publishes Peer Advertisements on behalf of the proxied sensor motes, thus being present within a peer group as any other normal peer from the other member's standpoint.

In order to ease the location of sensors with particular capabilities, the Virtual Peer's advertisement includes the following extra information:

- *Peer Type:* Specifies that this is actually a jxSensor Virtual Peer.
- *Capabilities:* A list with the available characteristics of the sensor mote.
  - Location: Position of the peer
  - Sensor list: The list of sensing capabilities that the mote controls, specifying each type and kind of operations that can be executed.
- Pipe: The Pipe Advertisement that will be used to contact this sensor mote, thus behaving just like any other JXTA service.

A sample Virtual Peer Advertisement is shown in Listing 1. The *Virtual Peer type* is included directly in the Advertisement's *Desc* field, using the "jxSensor" string. In this way, it is easy to search all the Virtual Peers inside a JXTA network, by looking up advertisements with this particular description. The *capabilities* and *pipe* information are included in the service parameter list, indexed using the *Svc* advertisement tag.

XML Listing 1 - Virtual Peer Advertisement

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE jxta:PA>
<jxta:PA xml:space="default" xmlns:jxta="http://jxta.org">
  <PID>urn:jxta:uuid-59616...7B03</PID>
  <GID>urn:jxta:uuid-425A5...5002</GID>
  <Name>sensorPeer_1</Name>
  <Desc>jxSensor</Desc>
  <Svc>
    <MCID>urn:jxta:uuid-DEADB...01105</MCID>
    <Parm>
      <jxta:PipeAdvertisement>
        <Id>urn:jxta:uuid-425A5...4704</Id>
        <Type>JxtaUnicast</Type>
        <Name>JXSENSOR-PIPE:sensorPeer_1</Name>
      </jxta:PipeAdvertisement>
    </Parm>
    <Parm>
      <jxSensor:Capabilities>
        <location>41.383333, 2.183333</location>
        <sensor>1,1,RWX</sensor>
        <sensor>2,1,RWX</sensor>
        <sensor>3,2,RWX</sensor>
      </jxSensor:Capabilities>
    </Parm>
  </Svc>
</jxta:PA>
```

The list of sensing capabilities a sensor mote controls is specified with an id, a descriptor code for its type and the permissions flags for the allowed operations. The id is used to identify a specific sensing capability in a sensor mote when a query is executed. We define the following code types: 1-Temperature, 2-Vibration, 3-Light, 4-Humidity, 5-Location, 6-Battery, and 7-Other.

The available permission flags are:

- R: Only historic data of this sensing capability, the one stored in the gateway, can be read.
- W: The configuration of the sensing capability can be modified.

- X: Data can be directly read from this sensing capability.

2) Sensor mote request data: The communication between a JXTA Peer and a Virtual Peer is done through a JXTA pipe, the one that can be found in the Virtual Peer Advertisement, as has been previously detailed in *sensor mote discovery* section. The Virtual Peer listens at its Pipe and receives messages. In general, the information received from a pipe is binary data, but in jxSensor we have decided to use JXTA Messages in order to improve compatibility and readability. JXTA Messages can be encoded as XML documents formed by name-value couples, under a namespace. Additionally to the information that can be added in a JXTA Message, it contains extra information that is internally needed by JXTA pipes. But for simplicity and readability, those information is not shown in examples messages bellow.

The *sensor mote request data* operation consists in querying data that has been captured from one sensing capability in a Virtual Peer. This data can be directly read from a sensor mote or obtained from the historical cache stored at the Virtual Peer.

In the first case, a query JXTA Message requesting a *read current data* action has to be sent to the Virtual Peer. An example of this message is shown in Listing 2. The *operation* tag specifies the type of operation which in this case is X. The *id* and *timeout* tags specify the sensor id and the maximum time the Virtual Peer is going to wait for the answer of the sensor mote before sending an error response. Furthermore, the *transaction id* has to be added in order to match the query with the response at a later time.

XML Listing 2 - read current data msg

```
<jxSensorMessageQuery>
  <operation> X </operation>
  <id> 3 </id>
  <timeout> 5 </timeout>
  <transactionId> 93872088 </transactionId>
</jxSensorMessageQuery>
```

In the second case, a query JXTA Message requesting a *read historic data* action has to be sent to the Virtual Peer. This message shares the same structure as the one before but with the difference that is a different operation, R. An optional tag *time* can be added specifying the time when the measure was taken. If not specified, the most recent stored data is sent.

Two possible kind of responses can be obtained from these queries: the requested data or an error. If the data could be accessed, it will be received in a message which contains the transaction id, the data from that sensor and the time when that data was measured. If an error occurs, a different message will be received with the transaction id, and the error. The possible errors are operation not allowed, timeout and no data. The first case happens when an operation is performed and we do not have the permissions to do so, whereas the second case when there is no data for that period of time, and the last case, when the response was not retrieved after the specified time.

3) Sensor mote apply configuration: The third operation consists in executing a command to change the configuration of a sensing capability from a sensor mote. In general, all sensing capabilities allow setting how often a measure is taken and sent to the gateway (in seconds), but other configurations can be set if implemented in the *MoteAdapter* service. The message sent is similar to the one of the *request data* operation but containing how often a data will be read from the sensor mote and sent to the WSNGateway. The answer is similar to the one of the previous operation, but only the time when the operation was done is returned. If an error is produced, the same error message is received.

B. Sensor mote layer protocols

The format of messages exchanged between the sensor motes and the gateway is defined by the Sensor App running on sensor motes, which can be different in each scenario. The MoteAdapter service is responsible of translating the message format to the operations defined by jxSensor, which are invoked on the MoteAbstraction component, decoupling the messages sent over the WSN and the ones of JXTA. This allows to use the most efficient implementation for each operation at any case. Thus, this subsection focuses in the message exchanges between the Mote Abstraction component and the MoteAdapter service. The operations that should be implemented at the MoteAdapter service are:

- **Associate** a sensor mote to a gateway, and join to some groups.
- **Request data** from a sensor mote.
- **Apply configuration** of a sensor mote.
- **Send periodic data** from a sensor mote.

1) Association: The first operation a sensor has to do to belong to the JXTA network is associating with a WSNGateway. In this operation the sensor should describe itself in order to allow the WSNGateway to spoof its identity. The information required is its location, capabilities and available sensors, and groups and permissions. This association can be done in two ways: initiated by the sensor mote, or manually set at the WSNGateway.

If the sensor mote sends an association message to the WSNGateway, the *MoteAdapter* service translates it and sends the message to the *Associator* component, which will perform the operation and answer with an ACK message. An example of an association message is shown in Listing 3. The first line is the id of the sensor mote. The second line is the location of that sensor mote. The next two lines are the groups the Virtual Peer will belong and the last three lines describe three sensing capabilities and their permission. The first element is the id of the sensing capability, the second is the type of sensing capability (see Section IV) and the rest are the permissions for each group. For instance, the first sensing capability is a temperature one, allowing all the operations for the group A but not allowing to change the configuration of the sensor from group B.

On the other hand, information can be directly set at the WSNGateway, without the need of a message exchange. This is achieved using a console terminal session command. This method allows two powerful features. The first one is that

a live WSN could be integrated to JXTA without modifying anything at the sensor motes. Only the gateway has to be modified with the JXTA software and jxSensor services and MoteAdapter service adapted for the internal WSN protocol. Then the information of the motes can be added manually to the WSNGateway by this association procedure, and the WSN is directly integrated into JXTA. The second interesting scenario is when you have a WSN network where there is just one way communication between sensor motes to the Gateway but not the opposite. It is possible to add sensor motes manually, only allowing "R" permission to its sensing capabilities. Then, the JXTA network can take advantage of the information of such sensor motes although there is not really a way to directly query them.

XML Listing 3 - Association missage

```
5438764863;
41.383333, 2.183333;
A,urn:jxta:uuid-425A5C703CD5454F9C03938A0D65BD5002;
B,urn:jxta:uuid-425A5C703CD5454F9A49857DBF78765F002;
1,1,RWX,RX;
2,1,RWX,RX;
3,2,RWX,RX;
```

2) Request data and apply configuration: The MoteAdapter service, whether it is permitted based on the sensing capabilities, can send two types of petitions to the sensor mote. A query for a sensing capability, or a command to modify the configuration. The messages sent in the JXTA level have been described in Sections IV-A2 and IV-A3. These messages are going to be exchanged between the *Mote Abstraction* component and the *MoteAdapter* service.

3) Send periodic information: Sensor Apps usually send periodic messages with data captured from the mote's capabilities. Under our architecture, this information is sent to the Mote Abstraction component. In this way, historic data can be queried from JXTA peers without sending additional messages to the motes, and therefore not increasing its energy consumption. The message format is similar as the one of association, the first line is the id of the sensor mote, the second line is the time when measures were taken and the last three lines contain the id of the sensing capability, and the captured data.

V. Conclusions

In this paper we have presented jxSensor, an abstraction layer that allows JXTA peers to interact with WSN's. Through this integration, it is possible to provide a feasible solution to some of the WSN limitations on regards to data processing and dissemination.

Our proposal uses novel approaches that set it apart from others. First of all, each sensor mote, or an aggregation of them, can be treated as a single peer within the context of a peer group, allowing JXTA peers to communicate to them transparently. This allows sensor motes to use the versatility offered by JXTA Peer Groups to segment the network, such as enabling sensor motes to limit their presence to particular groups or allow different actions depending on the groups. Peer groups are also the main gateway to deploying security in JXTA. Secondly, an abstraction layer is set in the gateway. This permits to incorporate deployed sensor motes into CPS's without modifying its software. Thirdly, two-way communication is permitted, allowing both reading data from the sensor motes and configuring them. Nevertheless, networks with one way communication (sensor mote to gateway) are also supported, which is very useful in smart metering scenarios. Fourth, the concept of Virtual Peer is created. This allows a group of sensor motes to be seen as a single peer in the JXTA network, which allows to integrate WSN with a great quantity of sensors without clogging up the JXTA network. Finally, historic data is stored in the gateway in order to backup the information sent from the sensor motes and also to reduce readings by the sensor motes, optimizing resource utilization.

Our future work includes finishing the jxSensor implementation and testing it in a real scenario. We have sensor motes installed in civil infrastructures, such as bridges, which monitor the healthiness of the infrastructure. Since the network is already deployed, our goal is to incorporate the sensor motes to a JXTA network without modifying them. The flexibility and abstraction defined in jxSensor, via the *MoteAdapter* service, allows this kind of adaptation. Therefore, just the gateway software has to be updated. Other future plans are the study on how to improve the association procedure to allow the usage of credentials and secure Peer Groups, the creation of an application to monitor sensor motes information from a mobile phone using JXME[11] and finding ways to replicate historic data between different gateways to improve its availability.

References

[1] M. Dohler, I. Vilajosana, X. Vilajosana, and J. Llosa, "Smart cities: An action plan", in *Smart City Expo & World Congress. Dec*, 2011.

[2] Sun Microsystems Inc., "JXTA v2.0 protocols specification", 2007, https://jxta-spec.dev.java.net/nonav/JXTAProtocols.html.

[3] smokindoug, "Cut and paste with JXTA (clipboard manager)", 2009, http://kenai.com/projects/candpjini.

[4] K. Matsuo, L. Barolli, J. Arnedo-Moreno, F. Xhafa, A. Koyama, and A. Durresi, "Experimental results and evaluation of smartbox stimulation device in a P2P e-learning system", 2009, pp. 37–44.

[5] W. Song, S. Kim, S. Seok, and D. Cho, "A peer-to-peer environment for monitoring multiple wireless sensor networks", in *Proceedings of 18th Internatonal Conference on Computer Communications and Networks, 2009*, 2009, pp. 1–6.

[6] S. Seok, N. Kim, D. Choi, and W. Song, "An Implementation of P2P System for Sharing Sensory Information", in *Management Enabling the Future Internet for Changing Business and New Computing Services*, vol. 5787 of *Lecture Notes in Computer Science*, pp. 191–200. 2009.

[7] S. Krco, D. Cleary, and D. Parker, "Enabling ubiquitous sensor networking over mobile networks through peer-to-peer overlay networking", *Computer Communications*, vol. 28, no. 13, pp. 1586 – 1601, 2005.

[8] B. Lim, K. Choi, and D. Shin, "A JXTA-based Architecture for Efficient and Adaptive Healthcare Services", in *Information Networking. Convergence in Broadband and Mobile Networking*, vol. 3391 of *Lecture Notes in Computer Science*, pp. 776–785. 2005.

[9] A. Antoniou, I. Chatzigiannakis, A. Kinalis, G. Mylonas, S. Nikoletseas, and A. Papageorgiou, "A peer-to-peer environment for monitoring multiple wireless sensor networks", in *Proceedings of the 2nd ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks*. 2007, pp. 132–135, ACM.

[10] PC Engines, "Alix3d3", 2007, http://pcengines.ch/alix3d3.htm.

[11] Sun Microsystems, "JXME", 2003, https://jxta-jxme.dev.java.net.