2010 13th International Conference on Network-Based Information Systems

Sensor-Cloud Infrastructure

Physical Sensor Management with Virtualized Sensors on Cloud Computing

Madoka Yuriyama Takayuki Kushida
IBM Research - Tokyo
IBM Japan, Ltd.
1623-14, Shimotsuruma, Yamato-shi, Kanagawa-ken 242-8502, Japan
e-mail: {yuriyama, kushida}@jp.ibm.com

***Abstract*—Increasing numbers of physical sensors are used for various purposes. Those physical sensors are usually used by their own applications. Because each application manages both of physical sensors and their sensor data exclusively, other applications cannot use the physical sensors in the different party easily. We propose a new infrastructure called Sensor-Cloud infrastructure which can manage physical sensors on IT infrastructure. The Sensor-Cloud Infrastructure virtualizes a physical sensor as a virtual sensor on the cloud computing. Dynamic grouped virtual sensors on cloud computing can be automatic provisioned when the users need them. The approach to enable the sensor management capability on cloud computing. Since the resource and capability of physical sensor devices is limited, the cloud computing on the IT infrastructure can be behalf of the sensor management such as availability and performance of physical sensors. This paper describes the design of Sensor-Cloud Infrastructure, the system architecture and the implementation.**

Keywords-component; Sensor Networks; Ubiquitous Computing; Cloud Computing

I. INTRODUCTION

There are increasing numbers of physical sensors used for various purposes. For example, global shipments of automotive Microelectromechanical Systems sensors are expected to nearly double from 2006 to 2012 according to iSuppli Corp [1]. They forecast worldwide automotive MEMS sensor shipments will grow to 935.7 million units in 2012. Those physical sensors are connected to their own IT systems. Only applications within the IT systems can freely use the physical sensors. Because each application manages both of physical sensors and sensor data, users, having no physical sensors, cannot use the physical sensors directly. If there were an infrastructure on which users could share multiple different kinds of physical sensors easily, many new services could be provided via this infrastructure.

Cloud computing services for IT resources provide users with virtual servers. Users can use the virtual servers with no concerns about the locations of the servers or their detailed specifications. We propose an infrastructure "Sensor-Cloud infrastructure" on which users can use the sensors without worrying about their locations and detailed specifications. Sensor-Cloud infrastructure virtualizes multiple physical sensors as "virtual sensor." For example, if there are many temperature physical sensors for each floor of a building, one virtual sensor could be defined for each floor (such as a virtual temperature sensor just for the 4th floor).

IT systems are managed by system management mechanisms. Existing studies of physical sensors focus on data processing [2], routing [3] [4], power management [5], clock synchronization [6] [7] [8], localization [9] [10], OS [11] [12], and programming [13] [14]. There are few studies focusing on the management of physical sensors because current physical sensors are closely linked to each application directly. The requirements for sensor management mechanisms have not been clarified. The users will be dissatisfied if they cannot use their sensors when the sensors are needed. Sensors should be managed by system management mechanisms if they are provided as a high quality service. We divided into sensor system management and sensor data management in this paper. Sensor-Cloud infrastructure provides sensor system management. If existing sensor data management studies use our sensor system management, the usability will be improved.

Our Sensor-Cloud infrastructure provides virtual sensors so the users need not worry about the real locations and the differences of multiple physical sensors. The users can use and control virtual sensors with standard functions. Dynamically grouped virtual sensors are provisioned automatically in response to the requests from users. Users can destroy their virtual sensors quickly when they become unnecessary. Monitoring virtual sensors is used to maintain the quality of service. Sensor-Cloud infrastructure also provides a user interface for registering or deleting of physical sensors, for requesting for provisioning or destroying virtual sensors, for controlling and monitoring virtual sensors, and for registering of deleting users. We discuss related work in Section 2. We present an overview of Sensor-Cloud infrastructure in Section 3. The system architecture appears in Section 4 and the details of our implementation are in Section 5. We discuss the pros and cons in Section 6 and conclude in Section 7.

II. RELATED WORKS

There have been a few of studies on the management of physical sensors. Users need to know the specifications of different kinds of physical sensors. OGC (Open Geospatial Consortium) [15] defined Sensor Modeling Language

978-0-7695-4167-9/10 $26.00 © 2010 IEEE
DOI 10.1109/NBiS.2010.32

(SensorML) [16] to provide standard models and an XML encoding for physical sensors' description and measurement processes. SensorML can represent the metadata for any physical sensor (such as the type of physical sensor, the location, and the accuracy). We use SensorML to describe the metadata of physical sensors. We added the mapping between physical sensors and virtual sensors to describe how to translate commands coming from users to virtual sensors into commands for the corresponding physical sensors.

Although there are many kinds of physical sensors, no application need use all of them. Each application needs sufficient physical sensors for its requirements (such as physical sensors in a certain location). A publish/subscribe mechanism [17] is used to select physical sensors in [18]. When there are multiple sensor networks, each sensor network publishes sensor data and metadata that describes the type of physical sensors. Each application subscribes to one or more sensor networks to receive a real-time data stream from their physical sensors. Such publish/subscribe mechanism allows each application to selects only the type of physical sensors it collects data from. Sensor-Cloud infrastructure makes virtual sensors from multiple physical sensors. Because every virtual sensor is not created from a sensor network, the grouping is more flexible. Users can select groups of virtual sensors or virtual sensors.

Users should check whether the physical sensors are available and detect physical sensors' faults for keeping the quality of the data coming from physical sensors. FIND [19] provided a novel method to detect physical sensors with data faults. FIND ranks the physical sensors based on their sensing readings as well as their physical distances from an event. FIND considers a physical sensors' faulty if there is a significant mismatch between the sensor data rank and the distance rank. This approach focuses on detecting physical sensors' faults, while we focus on monitoring the virtual sensors. Because there is a relationship between the status of a virtual sensor and the status of its sensors, the virtual sensor will also report incorrect results if the linked physical sensors are faulty. The users of the cloud computing service check the status of their virtual servers, not the status of the linked physical server. We also focus on monitoring the status of virtual sensors.

III. Overview of Sensor-Cloud Infrastructure

Fig. 1 is an overview of Sensor-Cloud infrastructure. The top part is the lifecycle of service delivery. Various sensors with different owners can join Sensor-Cloud infrastructure. Each owner registers or deletes its physical sensors. The templates for virtual sensors and virtual sensor groups are prepared for sharing physical sensors. Users request virtual sensors or virtual sensor groups by selecting templates, use their virtual sensors after provisioning and release them when they become unnecessary. They can access Sensor-Cloud infrastructure via the user interface on web browser.

When users request virtual sensors or virtual sensor groups, Sensor-Cloud infrastructure automatically provisions them from their templates. Users can control their virtual sensors directly or via their Web browsers. Sensor-Cloud infrastructure also provides the users with monitoring functions for the virtual sensors.

We first discuss the design points of Sensor-Cloud infrastructure in this section. Then we describe the various actors related to Sensor-Cloud infrastructure.

A. *Design Points*

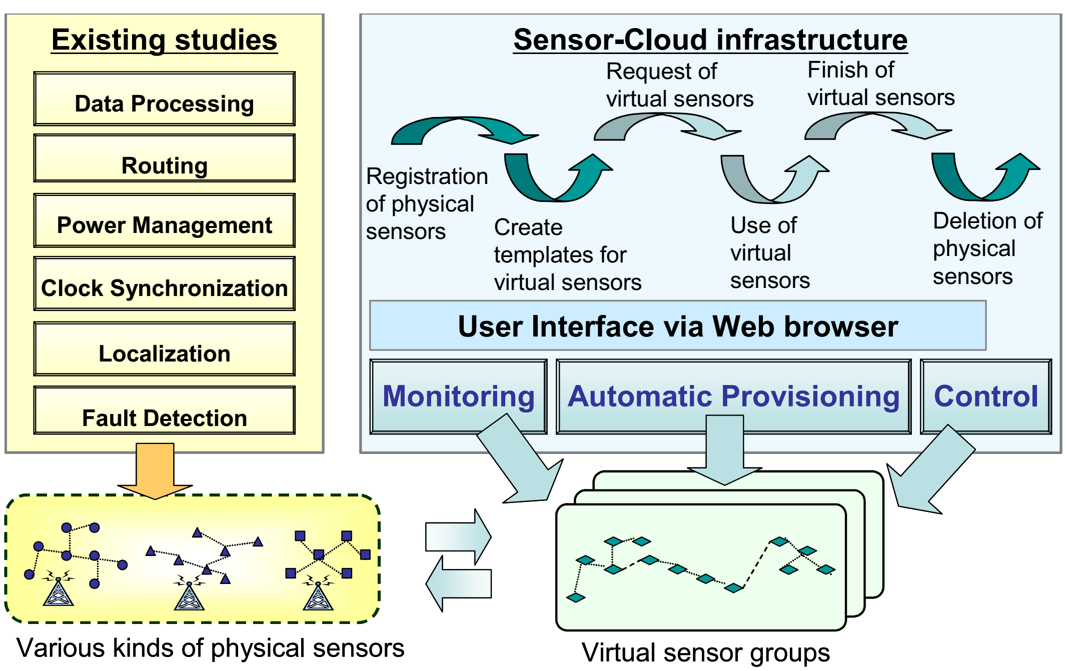

Figure 1. Overview of Sensor Cloud Infrastructure

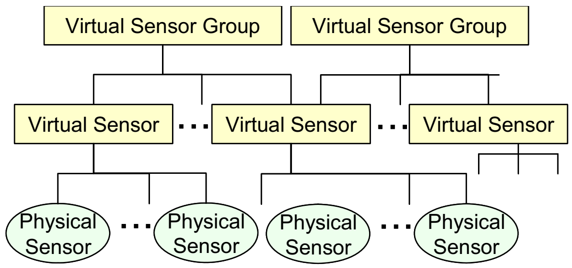

Figure 2. Relationship among Virtual Sensor Groups, Virtual Sensors, and Physical Sensors

There are many various physical sensors owned by different owners. When an application or middleware needs to use some sensors, the required sensors should be dynamically organized.

1) Virtualization: There are various kinds of scattered physical sensors. We propose virtual sensor and virtual sensor group in order for the users to be able to use sensors without worrying about the locations and the specifications of physical sensors. Fig.2 describes the relationship among virtual sensor groups, virtual sensors, and physical sensors. Each virtual sensor is created from one or more physical sensors. A virtual sensor group is created from one or more virtual sensors. Users can create virtual sensor groups and freely use the virtual sensors included the groups as if they owned sensors. For example, they can activate or inactivate their virtual sensors, check their status, and set the frequency of data collection from them,. If multiple users freely control the physical sensors, some inconsistent commands may be issued. The users can freely control their own virtual sensors by virtualizing the physical sensors as virtual sensors.

2) Standardization: Different kinds of physical sensors have different specifications. Each physical sensor provides its own functions for control and data collection. Standard mechanism enables users to access sensors without concern for the differences among the physical sensors. We define standard functions for virtual sensors, so the users can access the virtual sensors with the standardized functions. Sensor-Cloud infrastructure translates the standard functions for the virtual sensors into specific functions for the different kinds of physical sensors.

3) Automation: Automation improves the service delivery time and reduces the cost. If there are operations involving humans, those services will be slow and expensive. Sensor-Cloud infrastructure prepares templates for the specifications of various physical sensors. When users select the template of a virtual sensor or virtual sensor group, Sensor-Cloud infrastructure dynamically and automatically provisions the virtual sensors in that virtual sensor group from the templates. Sensor-Cloud infrastructure is an on demand service delivery and supports the full lifecycle of service delivery from the registration of physical sensors through creating templates, requesting of virtual sensors, provisioning, starting and finishing to use virtual sensors, and deleting the physical sensors. These forms of support are automatic and delivered without human operations.

4) Monitoring: Because the application has troubles if it cannot use the sensor data from the virtual sensors, the application owner should check whether or not the virtual sensors are available and monitor their status for sustaining the quality of the service. The users can check the status and the availability of the virtual sensors by the monitoring mechanism of Sensor-Cloud infrastructure

5) Grouping: Although there are many kinds of physical sensors, each application does not have to use all of them. Each application uses some types of sensors or when the sensors which match certain constraings (such as a location). Sensor-Cloud infrastructure can provide virtual sensors as virtual sensor groups. Users can control each virtual sensor and virtual sensor groups. For example, a user can set the access control and the frequency of data collection for virtual sensor groups. Sensor-Cloud infrastructure prepares typical virtual sensor groups and users can create new virtual sensor groups by selecting virtual sensors.

6) Service Model: When the physical sensors are only for a specialized application, that application can freely uses and manages its own physical sensors. Sensor-Cloud infrastructure provides the infrastructure to share various sensors as a service. Sensor-Cloud infrastructure is responsible for maintaining the quality of the service. We defines the roles assigned to the participants joining the service, considering their merits and creating an appropriate cost model to support the service. We define the participants in the service as actors and describe them in the next section.

B. Actors on Sensor-Cloud Infrastructure

Fig.3 shows the relationships among actors and Sensor-Cloud infrastructure.

1) Sensor Owner: A sensor owner is an actor who owns has physical sensors. A sensor owner allows others to use those physical sensors through Sensor-Cloud infrastructure. One of the possible advantages for sensor owner could be rental fees for using the physical sensors. The fees reflects the actual usage of the physical sensors. A sensor owner registers the physical sensors with their properties to Sensor-Cloud infrastructure. The owner deletes the registration of them when s/he quits sharing them.

2) Sensor-Cloud Administrator: The Sensor-Cloud Administrator is the actor who manages the Sensor-Cloud Infrastructure service. The administrator manages the IT resources for the virtual sensors, monitoring, and the user interfaces. The administrator also prepares the templates for the virtual sensors and for some typical virtual sensor

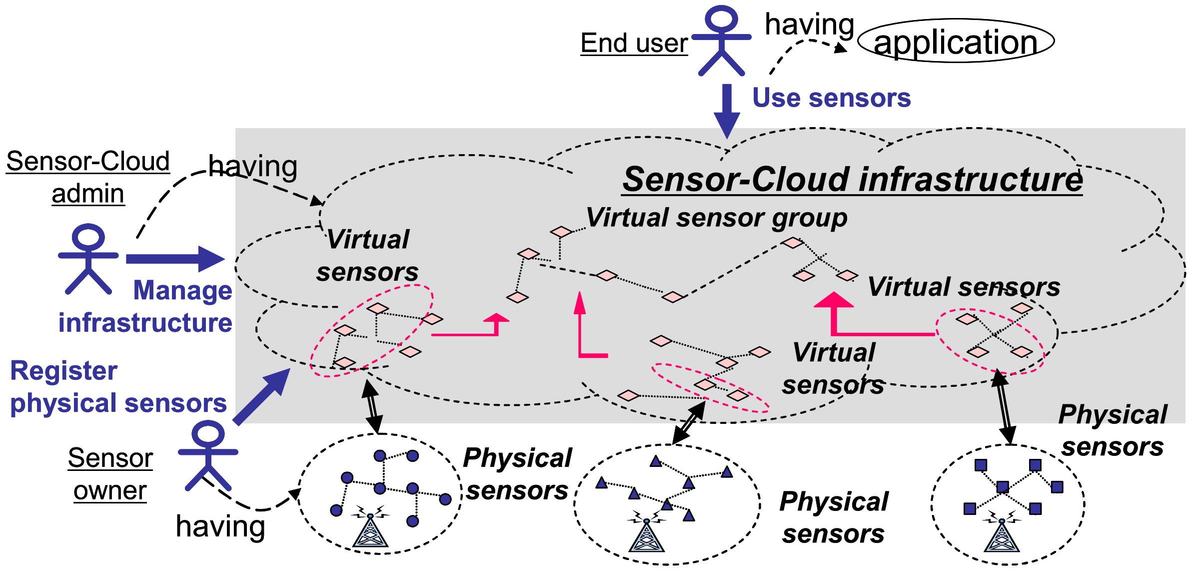

Figure 3. Relationship among Actors and Sensor Cloud Infrastructure

groups. The administrator can charge for the delivery of the Sensor-Cloud infrastructure service.

3) End User: An end user is an actor with one or more applications or services that use the sensor data. An end user requests the use of virtual sensors or virtual sensor groups that satisfy the requirements from the templates. The templates are prepared by Sensor-Cloud administrators. The user also can create a new template of virtual sensor group by selecting multiple templates of the virtual sensors or by modifying the existing template of the virtual sensor group or by their own active virtual sensors. The users can share their own templates among other end users. The user can control her/his virtual sensors directly or via a Web browser. The user can monitor the status of the virtual sensors. When they become unncessary, the user can release them. The end users can use the virtual sensors by paying for usage and with no detailed knowledge about the physical sensors.

We have defined the three kinds of actors according to the roles in Sensor-Cloud infrastructure. When the services of Sensor-Cloud infrastructure are delivered, the same person or organization may have the roles of both sensor owner and Sensor-Cloud administrator, especially when there are only a few kinds of physical sensors that are owned by an organization. The system is more scalable if Sensor-Cloud administrator is different from the sensor owners. Sensor-Cloud administrator focuses on the reliability and the quality of the service.

IV. System Architecture

Fig.4 shows the system architecture of Sensor-Cloud infrastructure. We divided it into the following seven main parts.

1) Client: Users can access the user interface of Sensor-Cloud infrastructure using their Web browsers.

2) Portal: Portal provides the user interface for Sensor-Cloud infrastructure.

3) Provisioning: Provisioning provides automatic provisioning of virtual sensor groups including virtual sensors.

4) Resource Management: Sensor-Cloud infrastructure uses IT resources for the virtual sensors and the templates for provisioning.

5) Monitoring: Sensor-Cloud infrastructure provides monitoring mechanisms.

6) Virtual Sensor Group: Sensor-Cloud infrastructure provisions virtual sensor groups for end users.

7) Sensors: Sensors are used in Sensor-Cloud infrastructure.

We first explain main components, and then we show the flows between components in this section.

A. Components

Here are the main functions of each component.

1) Portal server: When a user logs into the portal from a Web browser, the user's role (end user, sensor owner or Sensor-Cloud administrator) determinates the available operations. The portal server shows the end users the menus for logging in, logging out, requesting for provisioning or destroying virtual sensor groups, monitoring their virtual sensors, controlling them, creating templates of virtual sensor groups and checking their usage-related charges. The portal server gives sensor owners the menus for logging in, logging out, registering or deleting physical sensors, and checking the usage-related renatal fees. One of the menus for Sensor-Cloud administrators is for creating, modifying, and deleting the templates for virtual sensors or virtual sensor groups. Other menus are used to register or delete servers in the IT resource pool, to manage end users and sensor owners, and to check the status of IT resources. All of the menus for end users and sensor owners are available to Sensor-Cloud administrators as superusers. The portal server also sends requests to other servers as required.

2) Provisioning Server: The provisioning server provisions the virtual sensor groups for the requests from the portal server. It contains a workflow engine and predefined workflows. It executes the workflows in the proper order. First, it checks and reserves the IT resource pool when it receives a request for provisioning. It retrieves the templates of virtual sensors and virtual sensor groups, and then provisions the virtual sensor groups including virtual sensors on the existing or a new virtual server. After provisioning, the provisioning server updates the definitions of the virtual sensor groups. The virtual servers are provisioned with the agents for monitoring.

3) Virtual Sensor Group: A virtual sensor group is automatically provisioned on a virtual server by the provisioning server. Each virtual sensor group is owned by a end user and has one or more virtual sensors. The end user can control the virtual sensors. For example, they can activate or inactivate their virtual sensors, set the frequency of data collection from them, and check their status. If changing the configurations about the physical sensors is needed from the results from controlling the related virtual sensors, it acts for creating no conflict among the requests from other related virtual sensors. The virtual sensor groups are controlled directly or form a Web browser.

4) Monitoring Server: The monitoring server receives the data about virtual sensors from the agents in the virtual servers and the servers. It stores the received data in a database. The monitoring information for the virtual sensors is abailable using a Web browser. The Sensor-Cloud administrators are also able to monitor the status of the servers.

B. Comonent Flows

Here are the component flows for provisioning the virtual sensor groups.

1) Login: A end user logs in the portal on a Web browser.

2) Select the templates of virutal sensor group: The portal asks the database the list of the templates of virual sensors and virtual sensor groups. A end user selects the required templates from the list.

3) Request the virutal sensor group: A end user requests the virtual sensor groups by selecting the templates on the portal. The portal calls the provisioning server with the input parameters (such as the template IDs, the virtual group names, and user ID).

4) Reserve IT resource: The provisioning server first try to reserve the IT resource for the virtual sensor groups. If there is no spare resource on the existing virtual servers, it automatically provisions a new virtual server with a monitoring agent, and reserves the IT resource.

5) Get the templates and Provision: The provisioning server gets the templates of the virtual sensor group and the virtual sensors from the repository. It provisions the virtual

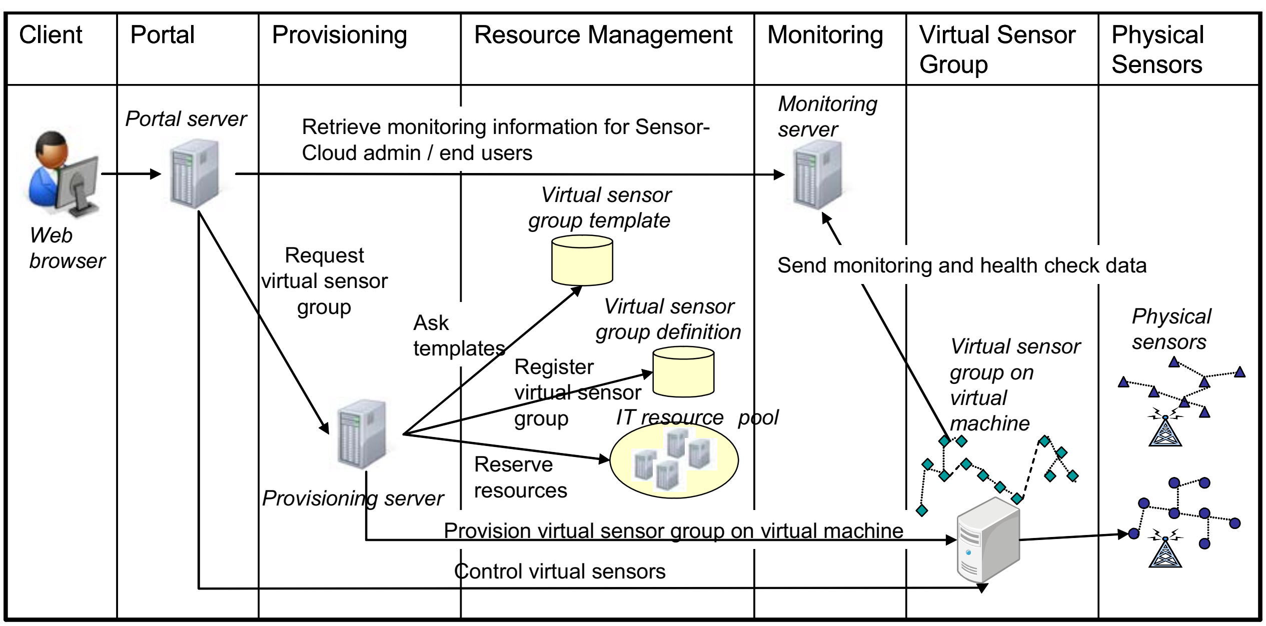

Figure 4. System Architecture of Sensor Cloud Infrastructure

sensor groups on the selected virtual server.

6) Notify the completion: The provisioning server notifies the end user of the completion of provisioning the requested virtual sensor group by e-mail. It also adds the new records to the definition of the virtual sensor groups.

Because there are no human operation after requesting, end users can start to use virtual sensors quickly. Automation improves the usability for end users and reduces the labor costs.

V. IMPLEMENTATION AND DEVELOPMENT

We developed a prototype of Sensor-Cloud infrastructure services.

A. Setup

Fig. 5 shows the current implementation. We first set up the two servers. Some virtual servers were automatically provisioned on the server for virtualization.

We constructed a management server which has the both roles of the portal server and the provisioning server. This management server also has a database and a repository. The database stores the definitions of the physical sensors, the provisioned virtual sensor groups, and IT resource. The definitions of the physical sensors describe the physical sensors' properties (such as sensor owner's ID, the kind of sensor data and sensor device data source's ID). The definition of virtual sensor group describes the provisioned virtual sensor group (such as virtual sensor group's ID, end user ID, the virtual server's ID and the creation date). The definition of IT resource pool describes the data about the servers and the virtual servers (such as IP address, host name, spec, and usage). The repository stores the templates of the virtual sensors and the virtual sensor groups. The virtual sensor template contains Java library and the property file describing the data mapping rule and the sensor device data source class name. The virtual sensor group's template defines the links to the templates of the virtual sensors, the creator ID, and the description. The workflow engine has the workflows for each purpose (such as provisioning virtual sensor groups, registering physical sensors and controlling virtual sensors). We used Jython [20] for creating workflows. Jython is an implementation of the Python programming language written in Java. The monitoring server receives the data from the monitoring agents in the virtual servers and stores in the database.

Each virtual server has a monitoring agent, a virtual sensor manager, and one or more virtual sensor group objects. A virtual sensor group object has a controller and one or more virtual sensor objects. The virtual sensor group controller has methods for controlling virtual sensor objects (such as activating or inactivating them and set the frequency of data collection). The methods are called by the workflow for controlling virtual sensors. The end users' application can also call the methods directly. The virtual sensor object has a standard access data method and sensor device data source specification. The application can get sensor data by the standard access data method. The virtual sensor object accesses each physical sensor via sensor device data source class which is implemented according to the sensor device data source specification. We use Mica2 mote as physical sensors.

B. Provisioning Flow

Fig. 6 indicates the flow for provisioning a virtual sensor group.

1) Request: A end user logs in the portal on a Web browser (1). The portal server gets the list of templates of virtual sensors and virtual sensor groups from repository (2) and shows them to the user. The user request for selecting and provisioning a virtual sensor group. The portal server calls the workflow engine with received the user ID and the

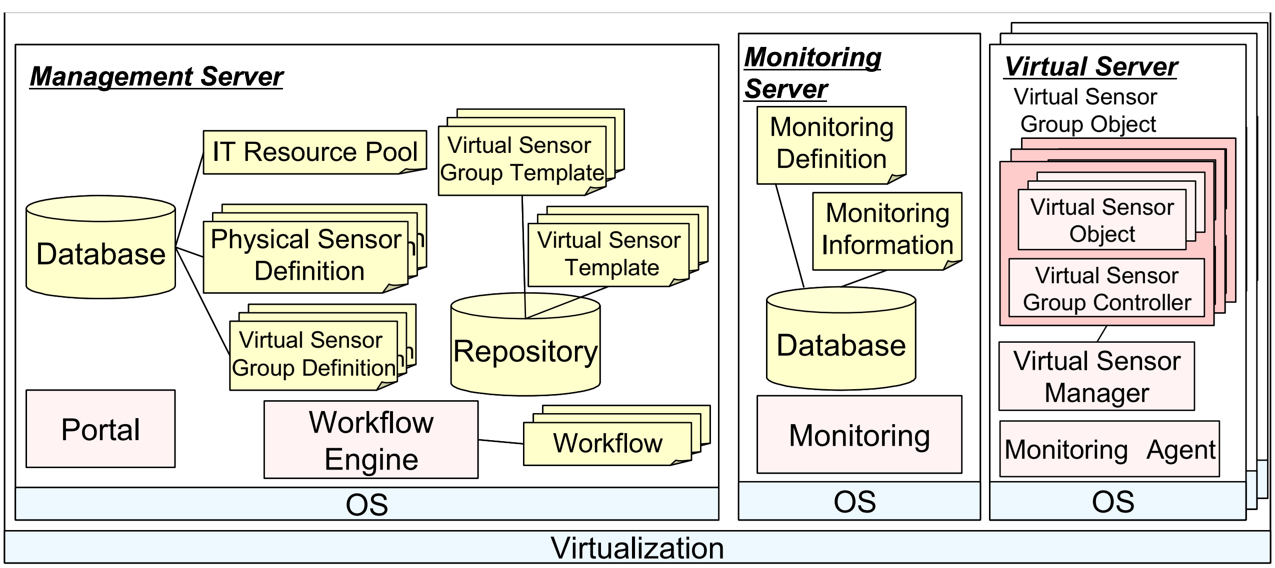

Figure 5. Prototype of Sensor Cloud Infrastructure

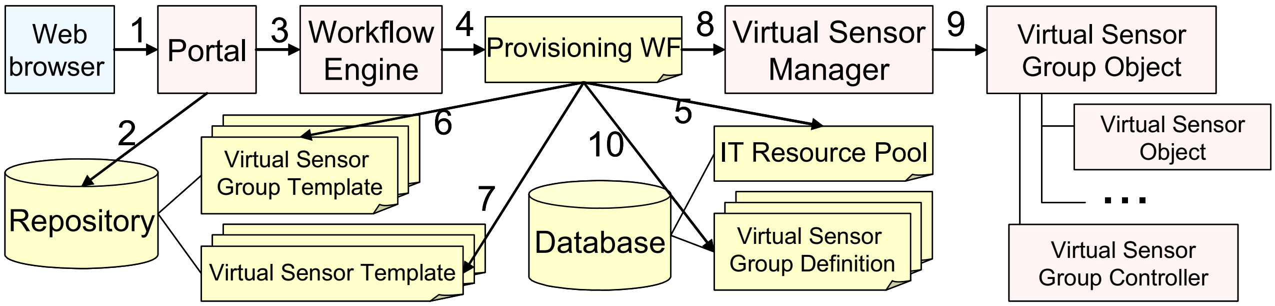

Figure 6. Provisioning Flow

requesting information such as the template ID of the required virtual sensor group and the name (3).

2) Call workflow engine: The workflow engine executes the workflow for provisioning virtual sensor groups (4). The workflow updates IT resource pool for reservation and gets the virtual server's information (5). It gets the template of the virtual sensor group template by ID (6) and the virtual sensors' templates from the links in the template of the virtual sensor group (7). The workflow sends the necessary Java library and the property files to the virtual server. It creates an instance of virtual sensor manager if there is no instance (8). It also creates an instance of virtual sensor group object, an instance of virtual sensor group controller, and one or more instances of virtual sensor object (9).

3) Post Provisioning: The workflow adds the definition of the new virtual sensor group to the database (10). Finally, it notifies the end user of the completion of provisioning

C. *Use cases*

We write the two use cases for describing the usability of Sensor-Cloud infrastructure.

1) Weather Service: When a typhoon is coming from the west area (such as from "Shikoku" to "Kinki" and "Hokuriku"), it is expected that the heavy rain occurs suddenly. The end user having a weather service requests Sensor-Cloud infrastructure to construct a virtual sensor group including the required virtual sensors (such as rainfall, water level, and traffic in these areas). The weather service will be able to use the virtual sensors, analyze multiple sensors' data timely, give rapid warning, and control traffic effectively. If some physical sensors are destroyed by the typhoon, the user can know the unreachable problem by the monitoring mechanism. When the typhoon goes away, the user deletes the virtual sensors. This use case shows the usability about rapid provisioning and deleting virtual sensors and monitoring mechanism.

2) Hospital Service: Several kinds of sensors are used in the hospital (such as heart rate sensor and sensor for oxygen saturation in the blood). These sensors are used by the hospital services for supporting daily medical cares, for improving the service level, and for preventing mistakes. Another emergency hospital service prepared the virtual sensors and inactivated them. When a disaster occurs, a lot of patients are suddenly carried to the hospital. The emergency hospital service turns the virtual sensors on, receives the sensor data, and support the doctors. This use case indicates the control mechanism of the virtual sensors which inactivates the virtual sensors when they are not needed and turns the virtual sensors on quickly when they are necessary. This use case also describes sharing sensors among multiple services.

VI. DISCUSSIONS

Sensor-Cloud infrastructure virtualizes sensors and provides the management mechanism for virtualized sensors. We compare (1) Sensor-Cloud infrastructure with (2) direct sharing physical sensors. Each application connects to multiple physical sensor networks directly and uses the physical sensors in direct sharing physical sensors. Table 1 shows the pros and cons. End users can use sensors without worrying about the details (such as location and specification) by virtual sensors on Sensor-Cloud infrastructure. End users can control their virtual sensors and monitor the status of them by the management mechanism on Sensor-Cloud infrastructure. Automatic provisioning enables end users to use the virtual sensors quickly and to release them when they become unnecessary. Virtual sensor groups provides dynamic grouping of sensors. On the other hand, direct sharing physical sensors does not have to prepare IT resource and predefined templates which Sensor-Cloud infrastructure needs. End users can neither check the status of sensors nor select the sensors dynamically by direct sharing physical sensors.

Each end user owns the virtual sensors in Sensor-Cloud infrastructure. If Sensor-Cloud infrastructure provides grouping of end users, the end users can share their virtual sensors by other users. All end users can see the templates of virtual sensor groups and virtual sensors defined by Sensor-Cloud administrators. If Sensor-Cloud infrastructure has an isolation mechanism, Sensor-Cloud administrators can configure some templates for the restricted users. We continue to design the detail of grouping and isolation.

VII. CONCLUSION

We present Sensor-Cloud infrastructure which virtualizes physical sensors in order for end users to share them with no

TABLE 1. PROS AND CONS OF SENSOR-CLOUD INFRASTRUCTURE AND DIRECT SHARING PHYSICAL SENSORS

	Pros	Cons
(1) Sensor-Cloud infrastructure	● End users can use sensors without worring about the details. ● End users can control their virtual sensors freely. ● End users can monitor the status of their virtual sensors. ● End users can start to use the virtual sensors quickly by automatic provisioning and release them when they become unnecessary. ● End users can create the group of sensors dynamically by virtual sensor groups ● Sensor owner can check the usage of their physical sensors.	● Sensor-Cloud infrastructure should prepare IT resource. ● Sensor-Cloud administoratos have to prepare the templates for virtual sensors.
(2) direct sharing physical sensors	● Direct sharing physical sensors does not have to prepare IT resource or the templates..	● End users cannot check the status of the sensors. ● End users should know the details of the senosrs.. ● End uses cannot select the sensors dynamically. ● End users cannnot use the sensors only during the sensors are needed.

concerns about the details of them (i.e. location and specification). Sensor-Cloud infrastructure enables end users to create virtual sensor groups dynamically by selecting the templates of virtual sensors or virtual sensor groups with IT resources. Because the end user can also create a new template of virtual sensor group, the user can create a virtual sensor group more flexibly. Users can use virtual sensors only during needed by the automatic provisioning mechanism. Automatic provisioning also improves the delivery time, reduces the cost and increases the usability of the users. The end users can check the status of their virtual sensors by the monitoring mechanism. They can also control their virtual sensors freely.

We divided into sensor system management and sensor data management. Sensor-Cloud infrastructure focuses on sensor system management. If existing or new sensor data management studies integrate our sensor system management, they will be able to provide higher quality services.

REFERENCES

[1] Automotive MEMS Sensor Market to Nearly Double by 2012 – iSuppli. http://www.isuppli.com/News/Pages/Automotive-MEMS-Sensor-Market-to-Nearly-Double-by-2012.aspx?

[2] S. R. Madden and M. J. Franklin, "Fjording the Stream : An Architecture for Queries Over Streaming Sensor Data," The 18th International Conference on Data Engineering, 2002.

[3] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss and P. Levis, "Collection Tree Protocol," The 7th ACM Conference on Embedded Networked Sensor Systems (SenSys 2009), 2009.

[4] M. H. Alizai, O. Landsiedel, J. A. B. Link, S. Goetz and K. Wehrle, "Bursty Traffic over Bursty Links," The 7th ACM Conference on Embedded Networked Sensor Systems (SenSys 2009), 2009.

[5] R. Katsuma, Y. Murata, N. Shibata, K. Yasumoto and M. Ito, "Extending k-Coverage Lifetime of Wireless Sensor Networks Using Mobile Sensor Nodes," The 5th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob’2009), 2009.

[6] A. Rowe, V. Gupta and R. Rajkumar, "Low-power clock synchronization using electromagnetic energy radiating from AC power lines," The 7th ACM Conference on Embedded Networked Sensor Systems (SenSys 2009), 2009.

[7] C. Lenzen, P. Sommer and R. Wattenhofer, „Optimal Clock Synchronization in Networks," The 7th ACM Conference on Embedded Networked Sensor Systems (SenSys 2009), 2009.

[8] J. Koo, R. K. Panta, S. Bagchi and L. Montestruque, "A Tale of Two Synchronizing Clocks," The 7th ACM Conference on Embedded Networked Sensor Systems (SenSys 2009), 2009.

[9] K. Matsumoto, R. Katsuma, N. Shibata, K. Yasumoto and M. Ito, "Extended Abstract: Minimizing Localization Cost with Mobile Anchor in Underwater Sensor Networks," The Fourth ACM International Workshop on UnderWater Networks (WUWNet), 2009.

[10] Z. Zhong and T. He, "Achieving Range-Free Localization Beyond Connectivity," The 7th ACM Conference on Embedded Networked Sensor Systems (SenSys 2009), 2009.

[11] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler and K. Pister, "System architecture directions for networked sensors," International Conference on Architectural Support for Programming Languages and Operating Systems, 2000.

[12] K. Klues, C. Liang, J. Paek, R. Musaloiu-E, P. Levis, A. Terzis and R. Govindan, "TOSThreads: Thread-Safe and Non-Invasive Preemption in TinyOS," The 7th ACM Conference on Embedded Networked Sensor Systems (SenSys 2009), 2009.

[13] T. I. Sookoor, T. W. Hnat, P. Hooimeijer, W. Weimer and K. Whitehouse, "Macrodebugging: Global Views of Distributed Program Execution," The 7th ACM Conference on Embedded Networked Sensor Systems (SenSys 2009), 2009.

[14] J. S. Miller, P. Dinda and R. Dick, "Evaluating A BASIC Approach To Sensor Network Node Programming," The 7th ACM Conference on Embedded Networked Sensor Systems (SenSys 2009), 2009.

[15] Open Geospatial Consortium. http://www.opengeospatial.org/

[16] SensorML. http://vast.uah.edu/SensorML/

[17] J. Shneidman, P. Pietzuch, J. Ledlie, M. Roussopoulos, M. Seltzer and M. Welsh, "Hourglass: An Infrastructure for Connecting Sensor Networks and Applications," Harvard Technical Report TR-21-04, 2004.

[18] M. Gaynor, M. Welsh, S. Moulton, A. Rowan, E. LaCombe, and J. Wynne, "Integrating Wireless Sensor Networks with the Grid," IEEE Internet Computing, Special Issue on Wireless Grids, 2004.

[19] S. Guo, Z. Zhong and T. He, "FIND: Faulty Node Detection for Wireless Sensor Networks," Proc. The 7th ACM Conference on Embedded Networked Sensor Systems (SenSys 2009), 2009, pp. 253-266.

[20] The Jython Project. http://www.jython.org/..