Practical RFID + SENSOR Convergence Toward Context-Aware X-Reality

Marie Kim
RFID/USN Research Division, ETRI
138 Gajeongno, Yuseong-gu
Daejeon, KOREA
+82-42-860-1590
mariekim@etri.re.kr

Hwang Jae Gak
RFID/USN Research Division, ETRI
138 Gajeongno, Yuseong-gu
Daejeon, KOREA
+82-42-860-5251
jghwang@etri.re.kr

Cheol Sig Pyo
RFID/USN Research Division, ETRI
138 Gajeongno, Yuseong-gu
Daejeon, KOREA
+82-42-860-4929
cspyo@etri.re.kr

ABSTRACT
Second Life is a representative virtual world accessible via the Internet. One can live in Second Life as a Resident, represented as an avatar. Now an important question is a raise that is, What if virtual world reflects real world (almost) exactly in the same way and vice versa? RFID and sensor technologies can drive life into virtual world i.e., RFID + Sensor technology augments user's experiences in the virtual world and enables real world synchronous response toward the virtual world. Furthermore, these two worlds can cross over to extend human life space (called *cross-reality*). However, the main challenge of using RFID + Sensor technology in real world is the heterogeneity of RFID + Sensor devices and the sensing data. Ubiquitous Sensor Network (USN) middleware has emerged as the key technology to propel the innovative paradigm (i.e., cross-reality) realization because USN middleware provides solution to heterogeneity and more. This paper describes USN middleware (COSMOS) which intermediates between various USN applications and diverse physical resources such as RFID reader networks, sensor and actuator networks, etc. and shows how it functions as a part of cross-reality platform.

Categories and Subject Descriptors
C.2.4 [Computer-Communication Networks]: Distributed System

General Terms
Management, Design

Keywords
USN, RFID, Sensor, Actuator, X-reality, Middleware.

1. INTRODUCTION
Cross-reality (known as X-reality) is an informational or media exchange between real- and virtual-world systems [1]. The important point of X-reality is a conceptual paradigm shift from single-directional information flows to bidirectional information flows between two worlds. The differential characteristic of X-reality is that it can augment user's engagement in the experiences of virtual presence and virtual world. Ultimately, it results in the human life span extension from the only real world to both worlds. This innovative technology i.e., X-reality, needs some tools to deliver information (including media) or controls from virtual world to the real world and vice versa. USN (short for Ubiquitous Sensor Network) has emerged, as the technology to exchange information between the two worlds, synchronously and asynchronously. MIT (Massachusetts Institute of Technology) has been studying about using sensors as input to virtual world from real world [1]. While, video game device Wii manufactured by Nintendo [2] is a good example for using sensors to collect necessary data from physical world and to use them for virtual world story progression. Movie theater system, which uses various sensory effect devices, provides augmented contents to the audience by using sensory effect devices for audiences' immersion in the contents. Likewise, various sensors and actuators contribute to communication between two worlds, as inputs as well as outputs. By definition, USN is the technological infrastructure that operates and uses various sensor and/or actuators to provide intelligent context aware services to the human beings. Various kinds of sensors and actuators are developed and used in many different areas such as games, monitoring, public safety system, etc. Moreover, enhancement of wireless communication protocol makes sensor deployment easier and adoptable in real field services. With these technical supports and decreasing manufacturing costs, smart and diverse monitoring services are developed and utilized. For example, the air pollution monitoring service can use various types of sensors for measuring air pollutants, orientation of the wind, speed of wind, etc. Similarly, a remote healthcare service can use sensors for measuring body temperature, pulse, ECG, etc. In case of cold chain management service, it may use temperature sensors, humidity sensors, RFID readers and RFID tags for check delivery and/or storage environment for products. These types of services are called USN service because they use USN resources such as sensors, RFID readers, etc.

In reality, to implement and operate these all USN services, it is desirable to use a software platform between applications and devices to manage and interoperate various kinds of devices and data. This software platform may provide communication channels between various applications and diverse informational/controllable resources and provide useful functions to relieve application development and operation load. USN middleware provides such interfaces between USN applications and sensor networks, and offers other functions like resource discovery, query processing, resource monitoring, etc. for various USN applications [3]. By using USN middleware, USN applications (virtual world) can get environmental parameters' values (real world) and take certain actions toward physical environment (real world). At the same time, USN application can reflects the dynamics of physical environment based on the current sensor data collected by various sensor networks, RFID reader networks, etc.

2. REALTED WORKS

Related to USN technology, there are number of researches and standardization activities in various fields. These studies usually deals sensors, sensor data access, sensor device access, sensory effect device, sensory effect device access, sensor data processing, RFID tag data processing, lightweight communication protocols between sensor nodes, etc. Among them, MPEG-V, SWE (Sensor Web Enablement), USN middleware, and ALE (Application Level Events) deal with software framework to handles USN resources for supporting USN services.

ISO/IEC 23005 MPEG-V deals with interoperability between virtual world and real world, and between virtual worlds in terms of sensory information processing. ISO/IEC 23005 MPEG-V series includes overall system architecture for the MPEG-V framework, data scheme, etc. ISO/IEC 23005 MPEG-V architecture provides overall architecture for the interoperability between virtual world and real world and between virtual worlds. Figure 1 shows the MPEG-V architecture [4]. It aims at the interoperability between different worlds.

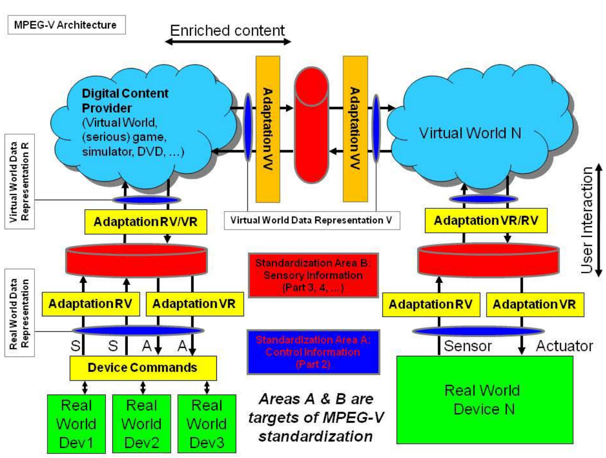

Figure 1: System Architecture of the MPEG-V Framework [4].

ISO/IEC 23005 Sensory Information provides the MPEG Representation of Sensory Effects (RoSE) [5]. This international standard specifies syntax and semantics of description schemes and descriptors that represent sensory information. This sensory information is then may be used to enhance the experience of users while consuming media resources. It describes with Sensory Effects Description Language (SEDL).

OGC (Open Geospatial Consortium) provides SWE (Sensor Web Enablement) framework. SWE consists of seven specifications. Those are web service interfaces (SOS, SPS, WNS, SAS) to access sensors and models and XML schema for sensors (SensorML), observations (O&M) and transducers (TML). The goal of SWE is to provide standard implementation framework to access observations collected from sensors for clients. Based on the analysis, it is expected that two standardization activities (ISO/IEC 23005, SWE) can compensate and collaborate with each other; ISO/IEC 23005 designs the framework for sensor effect device access and processing (virtual world → real world) while SWE designs the framework for sensor data (observations) access and processing (virtual world ← real world).

ITU-T makes overall USN service architecture standard (Y.USN-reqts) and USN middleware standard (F.usn-mw). F.usn-mw deals with USN service description and requirements for USN middleware and is at its final stage of the standardization process.

EPCglobal and ISO/IEC makes standards for RFID technology. ISO/IEC 18000-6 is the air interface standard between RFID reader and tag. Nowadays, a revision is in progress to embrace sensor tags. ISO/IEC 15961/15962 specify application interface and data encoding rules for RFID tags and ISO/IEC 24753 specifies data encoding and processing rules for sensors and batteries. EPCglobal ALE specifies an interface through which, clients may interact with filtered, consolidated EPC data and related data from a variety of sources.

Other than the above-mentioned researches, there are many researches related with USN resources such as IEEE1451.x series, Microsoft's SenseWeb, GSN (Global Sensor Network), etc. This paper describes USN middleware as a key solution for interoperation between virtual world (applications) and physical environment (real world). USN middleware is a software platform, which provides development and running environment for various types of applications. As a result, applications can use various USN devices (sensors, normal actuators, sensory effect devices, etc) via USN middleware.

3. REQUIREMENTS FOR X-REALITY

There is a need of various techniques to implement X-reality in real: resource discovery technique, resource identification technique, and exchange data model scheme, interface sets, query processing technique, etc. X-reality requires bidirectional information/control flows. More specifically, virtual world needs some situational data from the real world and requires taking certain actions toward virtual world. These VR (virtual-to-real world) directional operations require followings:

- Identify a list of necessary devices to access based on the virtual world contents.
- Check if necessary devices are available.
- Request data to appropriate devices.
- Request actions to appropriate devices.
- Reflect data and/or responses received from devices in the virtual world.

Conversely, real world needs some means to deliver data, responses and notifications processed by devices. These RV (real-to-virtual world) directional operations require followings:

- Process data, responses and notifications and deliver them to the exact clients, which request.
- Monitor and manage real world devices and report the status of them to clients.

USN middleware can provide necessary functions required by the X-reality realization except virtual world contents handling. That is a scope of the content processing technology. Section IV describes USN middleware technology. Through sections V and VI, this paper shows how USN middleware can support X-reality realization.

4. UBIQUITOUS SERVICE PLATFORM: USN MIDDLEWARE

USN middleware is a service platform, which provides necessary interface sets and functions for the USN applications and sensor networks [2]. The importance of USN middleware is that it makes USN applications implementation and sensor networks implementation decoupled; it divides USN industry into USN service provider area, USN middleware area and sensor network area. At the beginning, USN application implementation was highly dependent on the specifics of the sensor networks because they are developed in a tightly coupled way. This decoupling also makes sensor network implementation dependent on the applications' requirements. It enables sensor network providers to manufacture and to sell the same sensor networks for different service providers in a cost effective way. In one word, USN middleware is the key technology to make USN services and infrastructure practical. Figure 2 depicts the overall USN service framework.

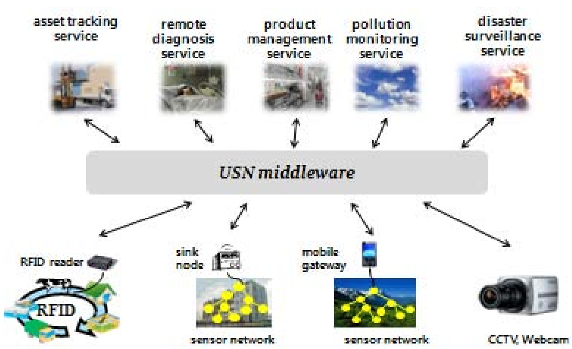

Figure 2: USN Service Framework.

The functions of USN middleware are derived from USN applications' common activities; such as query processing, sensor data integration, USN metadata directory service, sensor network monitoring, interfacing with sensor networks. In addition, intelligent USN middleware may provide intelligent data processing; sensor data mining, context-aware processing, event management, business brokering, etc optionally. Figure 3 shows the architecture of ETRI USN middleware (COSMOS).

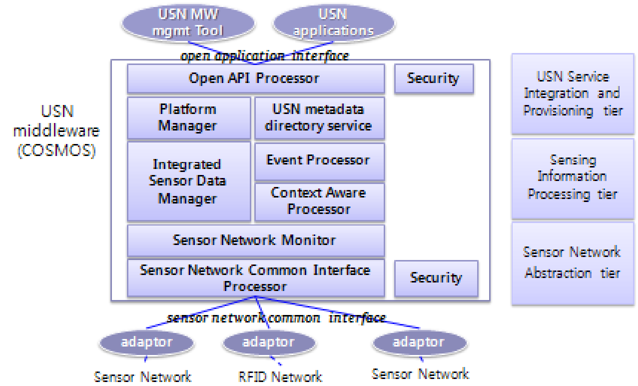

Figure 3: USN middleware architecture.

COSMOS consists of three tiers i.e., USN Service Integration and Provisioning tier, Sensing Information Processing tier and Sensor Network Abstraction tier. USN Service Integration and Provision tier includes Open API Processor, Platform Manager, USN metadata directory service and Security. Sensing Information Processing tier provides Integrated Sensor Data Management, Event Processor, and Context Aware Processor. Sensor Network Abstraction tier consists of Sensor Network Monitor, Sensor Network Common Interface Processor and Security.

The features of COSMOS are the Open API, Sensor network common interface, USN metadata directory service and Integrated Sensor Data Manager. COSMOS defines open application interface set and sensor network common interface set. Two interface sets realize the decoupling between applications and sensor networks. USN metadata directory service defines USN metadata schema and provides the values of USN static/dynamic metadata to applications and USN middleware functional components. Characteristic of this USN schema is such that, it has a hierarchical structure from transducer, sensor node to sensor network. The Integrated Sensor Data Manager makes optimal query plans for operating shared sensor networks and supports various types of query; instant, continuous, stream, event based on the sensor network capabilities registered at the USN metadata directory service. In addition, it supports logical sensor network concept. One or more physical sensor networks constitute one logical sensor network. It is desirable for sensor networks grouping and replacement flexibilities.

5. INTEGRATED DATA (RFID+SENSOR) PROCESSING: INPUT TO VIRTUAL WORLD

5.1 Request Real World to report situational data

Virtual world needs situational data from real world to synchronize with current real world environment. To get necessary data from real world, virtual world first check if necessary devices are deployed and available. Next, it needs to check what parameters are to be set to operate target devices. Then, virtual world issues commands for devices to collects data or to control. On receipt requests from virtual world, real world processes the requests (query/command) and generate reports for virtual world. USN middleware provides following functions for virtual world to real world operations:

- Open application interface for virtual world to submit queries.
- USN metadata directory services to look up necessary USN resource information. USN metadata directory service contains information of devices, supporting sensing types, sensor network topology, supporting query types, current metadata status, etc.
- Sensor data processing service such as Integrated Sensor Data Management, Event Processor and Context Aware Processor, for processing collected data and generation appropriate reports for virtual world.
- Sensor network common interface to command USN resources to do certain actions (actuators/sensors).

6. ENFORCING ACTIONS ON REAL WORLD: INPUT TO REAL WORLD

Virtual world needs to request some actions to real world according to the context of virtual world. These actions are categorized into two groups. One is for normal actuators and the other is for sensory effect devices.

6.1 Request Real World to take actions: Usual actuators

Based on the current context of virtual world, virtual world needs to activate and control some devices (actuators) for displaying (TV/Screen/etc), alarming (alarm/etc), sounding (speaker/mobile phone/etc), etc. Smart building monitoring application activates alarming devices when emergency happens. USN middleware also provides actuator-controlling functions. Actuators are attached to sensor nodes of sensor networks. Addressing scheme is {*sensor network identifier: sensor node identifier: transducer identifier*}. Transducer can be a sensor or actuator. Currently, three types of actuators are supported: i.e., On/Off, digital and text. On/off type actuator's operation is to on and off the device. Digital type actuator's operation is to change actuation value(s). For example, if light device is used, intensity of illumination is the value of operation. Text type actuator's operation is to display the text on the screen of actuator. If necessary, operation type and operation parameters can be extended.

6.2 Request Real World to take actions: Sensory effect devices

To augment user's virtual experience in real world, virtual world needs to activate and control some types of sensory effect devices. There are various types of sensory effect devices: such as light, flash, temperature, wind, vibration, water sprayer, scent, fog, shadow, etc [4]. Until recently, USN middleware does not consider this type of actuators. Therefore, actuator-handling module needs to be extended and modified to embrace these devices.

7. IMPLEMENTATION

USN middleware is developed and tested in various fields already. It is implemented with Spring and Java SE 1.5. Applications use SQL-like language to submit sensors/actuators queries and use xPath to retrieve USN metadata directory service. USN metadata directory service allows retrieve/add/delete/update operations according to their permissions. USN middleware communicates with sensor networks. It means default communication unit is a network. This network can be a usual wireless sensor network or a RFID reader network. If necessary, sensor node-unit access and transducer-unit access are possible. This is because USN middleware's target is to handle shared wide area multiple sensor networks for multiple monitoring applications. Therefore, it handles devices as network units by default. Due to network unit handling, it can access more sensor nodes than node-unit handling. Following TABLE 1 shows the simple comparison on node-unit-access versus network-unit-access.

TABLE 1: Total number of sensor nodes accessible comparison.

α: total number of sensor nodes to access
G: sensor network
G_i: i-th sensor network
N: number of sensor node within a sensor network
$N_i = \lvert G_i \rvert = \lvert \{n_1, n_2, \ldots, n_m\} \rvert = m$
n: number of connections which USN middleware keeps
When USN middleware accesses n sensor networks ($\{G_1, G_2, G_3, \ldots, G_n\}$), total number of sensor nodes is greater than number of connections which USN middleware keeps cleary.
$\alpha = \Sigma \mathbf{N_i} > \mathbf{n}$ ($0 < i < n+1$, i: integer)

TABLE 2 shows a faction of sensor network metadata structure. At the end of sensor network description, sensor node specifications, transducer specification, supported sensing type specifications follow. Sensor network 1225286574058799360 uses binary-message-type sensor network common interface on TCP and supports Aggregation/Continuous/Instant queries. It uses ID/PASSWORD-based authentication, message level encryption/decryption and use secure socket TLS. One of sensor node's identifier is "1297344169133254656".

TABLE 2: Sensor network metadata (partial).


```
<SensorNetwork id="1225286574058799360" connected="false">
  <Description>
    <Name>U-SilverCare</Name>
    <Location>ETRI7?</Location>
    <Function>healthcare</Function>
    <Manager>etri</Manager>
    <AirInterface>IEEE802.15.4</AirInterface>
     <FinishReport>true</FinishReport>
    <DataDeliveryMode>PULL</DataDeliveryMode>
   <LargeDataLocation>file:///temp/picture</LargeDataLocation>
   <CommonMessage>
      <ContentType>BINARY</ContentType>
     <Version>1.0</Version>
   </CommonMessage>
   <TransportProtocol>TCP</TransportProtocol>
   <.....>
   <QueryProcessing>
     <RequestCountAtOnce>5</RequestCountAtOnce>
     <DataBufferingTime>0</DataBufferingTime>
     <SupportingQueries>
       <Query>AggregationQuery</Query>
       <Query>ContinuousQuery</Query>
       <Query>InstantQuery</Query>
       <Query>EventQuery</Query>
     </SupportingQueries>
   </QueryProcessing>
   <Monitoring>
     <Period>1000</Period>
        <Parameters>
       <Parameter>NodeActiveStatus</Parameter>
       <Parameter>ParentNodeID</Parameter>
       </Parameters>
```

```
        </Monitoring>
        <Security>
          <Authentication>
             <Type>ID/PASSWORD</Type>
            <Password>1111</Password>
           <CertificateStoragePath>client-cert1.der</CertificateStoragePath>
          </Authentication>
            <Confidentiality>
               <MessageEncryption use="true">
                   <Targets>
                       <Target algorithm="3DES/OFB">COMMAND</Target>
                            <Target
algorithm="3DES/OFB">NONSENSINGDATA</Target>
                            <Target
algorithm="3DES/OFB">SENSINGDATA</Target>
                    </Targets>
               </MessageEncryption>
               <UseTLS>false</UseTLS>
            </Confidentiality>
          </Security>
    </Description>
<SensorNodes>
    <SensorNode id="1297344169133254656" isActive="true">
         <Description>
         <....>
```

In this case, sensor network 1225286574058799360 has following capabilities: EventQuery processing and multiple queries processing. First, it supports EventQuery.,includes condition(s) to report sensing data. Sensor network keeps sensing according to its sampling mechanism but reports sensing data only if the specified conditions are met. For example, if EventQuery (select temperature from 1225286574058799360 WHERE temperature > 25) is sent to the sensor network, then the sensor network will report sensing data when sensed data value is larger than 30. This query is more effective than ContinuousQuery (select temperature from 1225286574058799360 period 30 for 2000) in terms of network traffics for sending reports. While, sensor network 1225286574058799360 can process maximum 5 queries at the same time. Element <RequestCountAtOnce> specifies the maximum number of queries processed at the same time within the sensor network. Following TABLE 3 shows the effectiveness of parallel queries processing.

TABLE 3: Processing time comparison (serial vs. parallel).

β: total application queries processing time

n: number of queries

$T_{query\_processing}(i)$: processing time for query i

$T_{query\_processing}L$: $T_{query\_processing}(i)$, 0< i< n+1, the least processing time for a query

In case of serial processing:

- $\beta_s = \Sigma\, T_{query\_processing}(i)$: $0 < i < n+1$, i : integer
- $\beta_s > (T_{query\_processing}L \times n)$ ..... **(1)**

In case of parallel processing:

- $\beta_p = T_{query\_processing}L + \delta$, δ: processing delay for parallel processing ....**(2)**

Obviously, (1) > (2) is true. Virtual world needs to check whether there are necessary resources, which it has to use. Virtual world submits retrieve-query to USN metadata directory service using xpath query. Then virtual world generates appropriate queries to access USN resources (sensors/actuators) and submits them to USN middleware. Format of query is SQL-like and TABLE 4 shows some exemplary queries.

TABLE 4: Exemplary quires.

Queries
select * from 1225286574058799360 period 10 for 2000
SELECT none FROM 1225286574058799360 where nodeId = 1297344169133254656 ACTUATION ACTUATOR 3675255604224800513 MODE ANALOG VALUE 15
SELECT none FROM 1225286574058799360 where nodeId = 1297344169133254656 ACTUATION ACTUATOR 3675255604213188609 MODE DIGITAL VALUE 0
SELECT none FROM 1225286574058799360 where nodeId = 1297344169133254656 ACTUATION ACTUATOR 3675255604234884609 MODE TEXT VALUE 'hello'
select jpg from 1225286574058799360 where nodeid=1297344169133254656

For simulation, COSMOS Test Driver made by ETRI submits these queries to USN middleware (COSMOS). Figure 4 shows the GUI of COSMOS Test Driver.

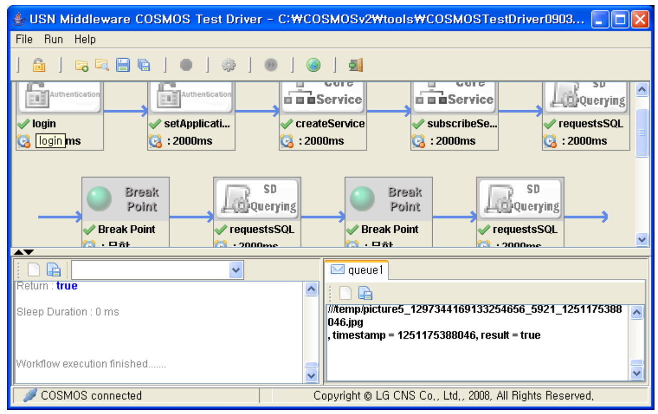

Figure 4: COSMOS Test Driver.

If virtual world submits query (select * from 1225286574058799360 period 10 for 2000) to COSMOS, then COSMOS replies sensing data collected by sensor network 1225286574058799360. TABLE 5 shows partial result from sensor network (1225286574058799360):

TABLE 5: An instance of results (select * from 1225286574058799360 period 10 for 2000).

```
SensorRecord] requestID = 1, snID = 1225286574058799360
[NodeResult] nodeID = 1297344169182070530
                name=temperature, type = FLOAT, value = 24.389795
                name=volt_voltage, type = FLOAT, value = 20.542852
                name=percent_voltage, type = FLOAT, value = null
                name=pulse, type = INTEGER, value = 15
                name=momentum, type = SHORT, value = 16
                name=binary, type = BYTE, value = [B@19872ad
                name=electrocardiogram, type = SHORT, value = 18
                name=bodyheat, type = FLOAT, value = 26.830046
                name=jpg, type = LONG, value = null
                name=mp3, type = LONG, value = null
, timestamp = 1251175365859, result = true
```

If the sensing type is large volume data (larger than maximum size of a sensor network common interface message), then it saves the data at the predefined location and reports the location to virtual world:

TABLE 6: An instance of large volume sensing data.

[SensorRecord] requestID = 5, snID = 1225286574058799360
[NodeResult] nodeID = 1297344169133254656
name=jpg, type = LONG, value = **file:///temp/picture5_1297344169133254656_5921_1251175388046.jpg**
, timestamp = 1251175388046, result = true

If the virtual world needs both sensor data and RFID tag data, it can submit a composite query to COSMOS such as a (SELECT * from 1225286574058799360, 3531136571836389632). Sensor network 1225286574058799360 includes various sensors and sensor network 3531136571836389632 includes a RFID reader. Then the result of a query is like this (given in TABLE 7):

TABLE 7: A Cartesian product result (sensor+RFID; partial).

[NodeResult] nodeID = 0
name=temperature, type = FLOAT, value = 23.801971
name=volt_voltage, type = FLOAT, value = 65.114395
name=percent_voltage, type = FLOAT, value = null
name=pulse, type = INTEGER, value = 37
name=momentum, type = SHORT, value = 46
name=binary, type = BYTE, value = [B@19872ad
name=electrocardiogram, type = SHORT, value = 50
name=bodyheat, type = FLOAT, value = 27.098211
name=jpg, type = LONG, value = null
name=mp3, type = LONG, value = null
name=rfidtag, type = STRING, value = urn:epc:id:usdod:2S194.1
, timestamp = 1251178469484, result = true

Results are Cartesian products of two reports from each sensor network. This result is one report out of multiple Cartesian products and shows the sensor data and RFID tag data at the same time. Via USN middleware, virtual world can get integrated data from multiple USN resources.

If the virtual world intends to activate certain devices, then, it submits queries to activating actuators. On receiving the queries, sensor network processes those requests and responds accordingly. For simulation, ETRI Sensor Network Simulator simulates the actions of adaptor and sensor network. Figure 5 shows the processing results of actuators at the simulator.

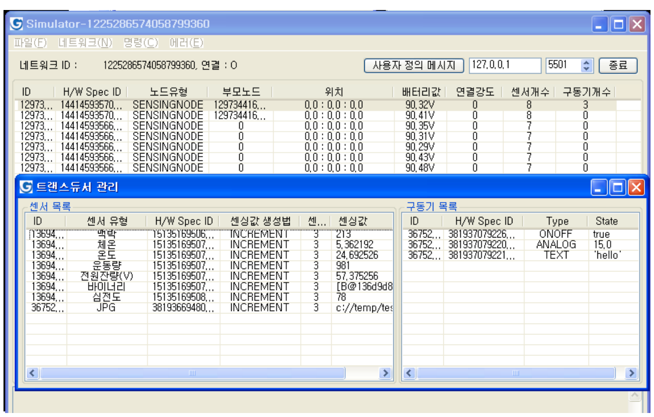

Figure 6: Simulator GUI.

Query (SELECT none FROM 1225286574058799360 where nodeId = 1297344169133254656 ACTUATION ACTUATOR 3675255604213188609 MODE DIGITAL VALUE 0) is sent to Simulator, which simulates sensor network (3531136571836389632) and adaptor, via COSMOS, and the action result is shown ((transducer) ID= 3675255604213188609, Type= ONOFF, State=true). It describes actuator (3675255604213188609) is an On/Off type actuator and it turns on.

The novelty of USN middleware is the new interface between USN middleware and sensor networks. New interface, called as Sensor Network Common Interface consists of connection-, query-, control-, channel control, and network info- interfaces. This interface set is scalable so that if a new interface is needed then it can be added easily. However, already existed interface should be considered first. TABLE 8 shows the interface message lists. Adaptor for the sensor network transforms the sensor network common interface to internal sensor network interfaces and vice versa.

TABLE 8: Sensor Network Common Interface.

Category	Messages
Connect/disconnect	ReqConnCtrl, ConnReqCtrl, ConnResCtrl, DisConnReqCtrl
Authentication	AuthReqCtrl, AuthResCtrl
Info request	NetworkInfoReq/Res, BufferDataReq/Res
Command control	CmdActionReq/Res, UpdateCmdReq/Res
Network/node control	ControlNetworkReq/Res, ControlNodeReq/Res
Sensing command	InstantCmd, ContinuousCmd, InstantEventCmd, InstantAggCmd, ContinuousAggCmd, RunActuatorCmd
Monitoring command	MonitoringStartCmd, MonitoringStopCmd
Report	SensingValueRpt, RunActuatorRpt, FinishRpt, SensingLargeValueRpt, MonitoringRpt, ErrorRpt, UpdateRpt
Channel chk	ChannelCheckCtrl, ChannelConfirmCtrl
User defined msg	UserDefinedMsg
NaK	NakChk

8. CONCLUSION

USN middleware provides useful functions for X-reality even though it needs some functional extension to support handling of sensory devices. . Via USN middleware, virtual world can get real world values by using sensors, RFID readers, etc and control devices in real world. Conversely, real world provides physical values to virtual world to synchronize two worlds. Therefore, USN middleware is a practical convergent solution for RFID + Sensor technology and thus for X-reality realization.

9. ACKNOWLEDGMENTS

This work was supported by the IT R&D program of MKE/KEIT [2006-S-022-01, Development of USN Middleware Platform Technology] and by a grant (#07KLSGC02) from Cutting-Edge Urban Development–Korean Land Spatialization Research Project funded by Ministry of Construction & Transportation of Korean government.

10. REFERENCES

[1] B. Coleman, Using Sensor Inputs to Affect Virtual and Real Environments,Pervasive Computing, 2009.

[2] http://www.nintendo.com

[3] M. Kim, J.W. Lee, Y.J. Lee and J.C. Ryou, “COSMOS: A Middleware for Integrated Processing over Heterogeneous Sensor Networks,” ETRI Journal, Vol.30. 2008, pp. 696-706.

[4] J.H.A. Gelissen, WD 2.0 of ISO/IEC 23005 MPEG-V, Architecture, April, 2009.

[5] C. Timmerer, S. Hasegawa and S.K. Kim, WD of ISO/IEC23005 Sensory Information, April, 2009.