ISO/IEC JTC 1 N9757

2009-09-17

Replaces:

ISO/IEC JTC 1
Information Technology

Document Type:	other (defined)
Document Title:	Standardization Issues for USN Middleware replying to JTC 1 Nara Resolution 31
Document Source:	SGSN Secretariat
Project Number:	
Document Status:	This document is forwarded to JTC 1 National Bodies for review and consideration at the October 2009 JTC 1 Plenary meeting in Tel Aviv.
Action ID:	ACT
Due Date:	
No. of Pages:	8

Secretariat, ISO/IEC JTC 1, American National Standards Institute, 25 West 43rd Street, New York, NY 10036; Telephone: 1 212 642 4932; Facsimile: 1 212 840 2298; Email: lrajchel@ansi.org

ISO/IEC JTC 1
Study Group on Sensor Networks

Document Number:	SGSN N151
Date:	2009-08-05
Replace:	
Document Type:	Other Document (Defined)
Document Title:	Standardization Issues for USN Middleware replying to JTC 1 Nara Resolution 31
Document Source:	Ad hoc group on Middleware
Document Status:	As per the SGSN Oslo Resolution 3, middleware issues will be reported to JTC 1 (Regarding the JTC 1 Nara Resolution 31).
Action ID:	FYI
Due Date;	
No. of Pages:	7

SGSN Convenor: Dr. Yongjin Kim, Modacom Co., Ltd (Email: cap@modacom.co.kr)
SGSN Secretary: Ms. Jooran Lee, Korean Standards Association (Email: jooran@kisi.or.kr)

1 Standardization Issues for USN Middleware

1.1 Definition of the Term Middleware

During the last years the number of USN applications has been increasing. Additionally the complexity of the application software on the nodes is much higher than it has been some years ago.
Today there is a large gap between the high level application requirements defined by the user and the complexity of the wireless sensor network itself.
Constrained energy and computing resources, the dynamics of the network topology and the low level application programmers interfaces of today's embedded operating systems require a lot of know-how which most application engineers don`t have.
Consequently the most **important task of a USN middleware is to bridge this gap**.

Definition:

Therefore the term "USN middleware" refers to software and tools which help to hide the complexity and heterogeneity of the underlying hardware and network platforms. Middleware has to ease the management of system resources and to increase the predictability of application executions.
USN middleware provides the services necessary for sensing-based ubiquitous computing applications which make use of a wireless sensor network and the related embedded operating system or firmware installed on the sensor nodes.
From today's point of view the following **tasks have to be fulfilled by USN middleware**:

- Support for application software engineer
- Abstraction of underlying hardware and software systems
- Provision of reusable codes or software services for software engineers
- Provision of monitoring services for quality of different services
- Maximize the resource efficiency for the whole USN system
- Warranty of data security and system robustness
- Support the integration of the USN into a existing IT landscape

The list is not comprehensive and has to be enhanced during the standardization activities. At this point it has to be mentioned that within the **scientific literature other definitions** can be found which put the focus on slightly different aspects. Due to restrictions in time and space all these definitions cannot be addressed in this document. They **have to be discussed and reviewed during the actual standardization process**. **Nevertheless**, from the SGSN's point of view the **definition presented above describes very well what a middleware for wireless sensor networks really is**.

1.2 Middleware in a System's Context

In order to understand the role and the concept of USN middleware comprehensively it is necessary to describe how it is embedded into an application context. Figure 2 gives some explanations.
The core of the figure is the wireless sensor network itself. Different topologies are possible. The network`s architecture can be hierarchical even from the application point of view. It is connected to the "rest of the world" via different types of gateways. One type might be mobile and integrated into a mobile phone or mobile computer, another one might be fixed to a part of the infrastructure. In the second case a backbone network like the internet or the communication infrastructure of a company is used for connecting the wireless network to a

smart object integration platform. The integration platform could be an extended “Smart Objects Integration Platform” which integrates additional functions or services that handle radio frequency identification tags or real time localization systems. It is self-evident that USN applications have to deliver benefits to the user. This user might be an individual consumer who wants to purchase a weather report using an internet service which is based on distributed sensor networks. He also might be a member of a company’s staff who is searching the World Wide Web in order to get location and state information for an important shipment using an internet tracking & tracing service based on a hierarchical network of sensor nodes attached to packages, parcels and containers. In both cases the user is following a well defined order to payment process which is embedded into an underlying service process that is carried out by the service provider.

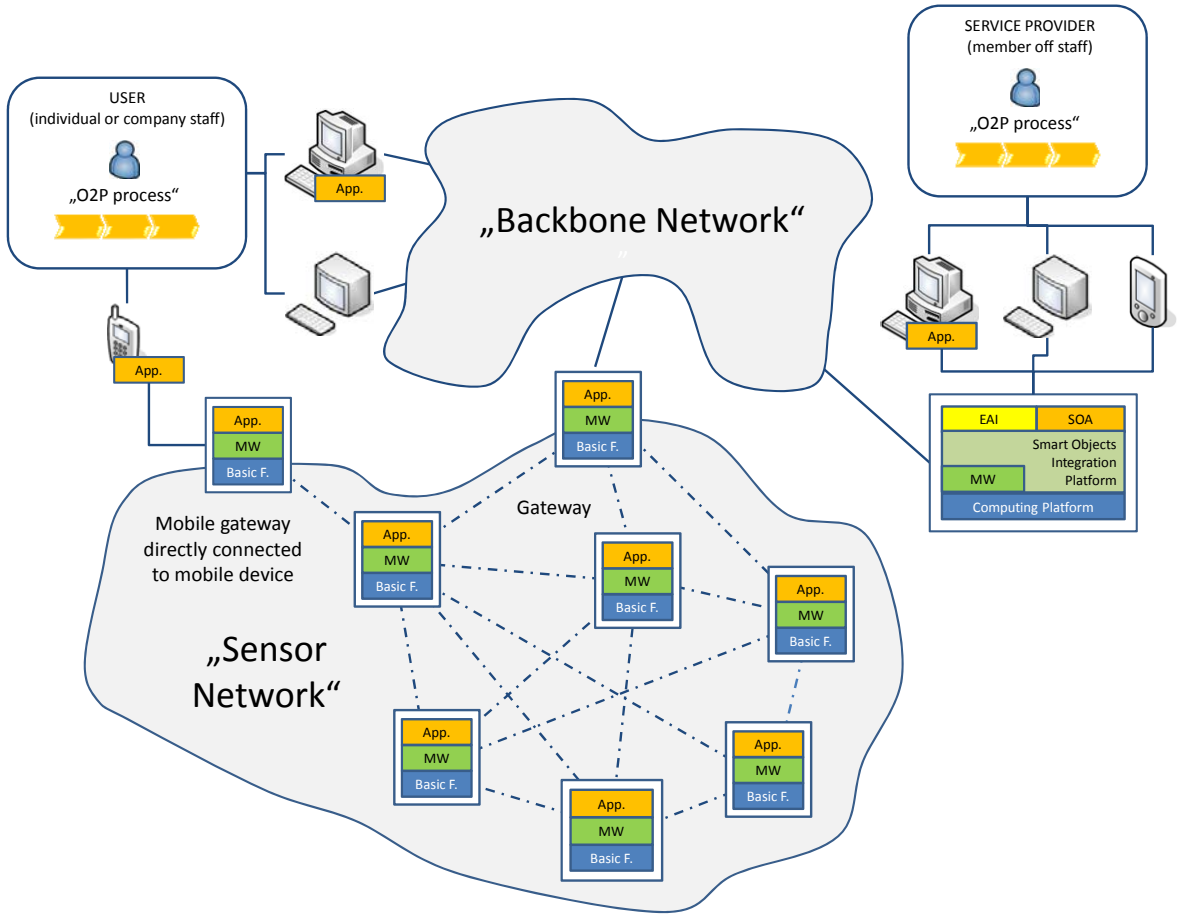

Figure 1: USN middleware embedded into an application context

Important is the fact that the application software used to fulfill the customer’s needs is not a monolithic block of code running on a central host computer. In most cases it will be distributed to mobile devices, fixed computers, USN gateways and sensor nodes which are interconnected using different types of network topologies.

It becomes quite clear that the USN middleware which has been presented in Figure 2 has to follow the same structure. It also has to be distributed.

The components of the USN middleware which are installed on nodes, gateways, phones and central host computers have to communicate and to work together in order to achieve some common goals and to support the distributed applications.

1.3 Analysis of existing Middleware Systems

A first scan of the existing scientific papers shows quite clearly that during the last years a lot of ideas and concepts for USN middleware have been developed. Nevertheless most of these papers focus on limited technological questions and singular applications. Only a few of them address more generic issues or try to summarize existing work in a structured and analytic manner.
Reference models which could be used as basic input to a standardization procedure are rare in literature. Figure 2 shows an example from [e.g. Miao-Miao Wang, Jian-Nong Cao and Sajal K. Das titled "Middleware for Wireless Sensor Nodes: A Survey" from May 2008].

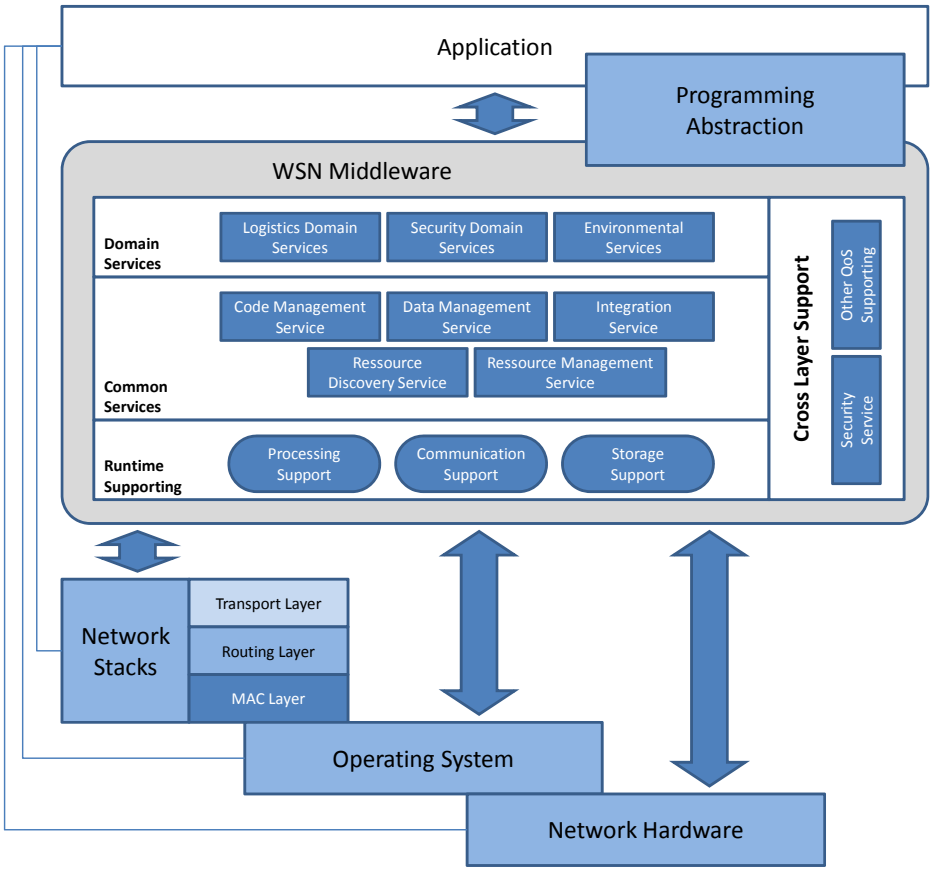

Figure 2: Example for a Reference model for USN middleware, according to [Wang/Cao/Das08]

According to the figure a typical middleware contains four mayor modules:

1. Programming abstraction: This module defines interface between Wireless Sensor Network middleware and the application software engineer.
2. System Services: These services can be used by the application software engineer in order to support the actual application.
3. Runtime Support: This module is an extension of the embedded operating system which supports the middleware services.[1]

[1] Runtime support is necessary because hardware and firmware are not always able to provide enough support for the implementation of the system services.

4. Quality of Service Mechanisms: These mechanisms have to cope with the fact that the quality of service restrictions often don't meet the requirements of a running application. They have to be regarded as an advanced feature of USN middleware and are still an open issue for research.[2]

The figure shows the complexity of the middleware problem. **Important is that there is no "one and only" middleware for wireless sensor nodes which meets requirements of all possible applications comprehensively.** During the last years many different solutions have been developed especially in research projects. **Members of the SGSN found more than 80 different prototypes or solutions**.
A short review of existing literature makes quite clear that a lot of design questions have to be addressed developing a USN middleware. For each question one can find different problem solutions. The following figure shows a morphologic box which gives a bird's eye view on the design problem. It could be used as an analytic means in order to describe existing solutions as well as a "tool box" for creating new ones. The first column represents the design questions, the matrix on the right hand side includes the possible answers.

Nr.	Design Question	Alternatives				
1	What is the principle of data collection?	continuous	query based	event driven		
2	How mobile is the application code executed on the nodes?	static (code does not move)	dynamic (code moves from node to node)			
3	On which level does programming abstraction take place?	node level	network or system level			
4	Which programming paradigm is used?	database paradigm	mobile agents	publish and subscribe	other paradigms	
5	What kind of interface is used to support the software engineer?	descriptive interface	imperative interface			
6	Which principle of code allocation on nodes is used?	node attributes	energy efficiency	QoS	security aspects	Other principles
7	Which mechanisms are used for code migration between nodes?	mobile agents	other concepts			
8	Which mechanisms are used for data acquisition?	acquisition based on events	query based date acquisition			
9	Which mechanisms are used for data processing?	centralized processing	node level processing	network level processing		
10	Which principle is used for data storage?	external storage	local storage	data centric storage		
11	What types of discovery services are provided by the middleware?	node resources	network topology	node localization	others	
12	What types of resource management services exist?	cluster service	schedule service	data routing service	others	
13	What types of backend information systems are supported?	database supported	EAI-based	Grid	others	

[2] Middleware acts as a broker between the application and the network infrastructure concerning QoS issues. It`s role might be the negotiation of new QoS guarantees in case that the required levels cannot be supported. It can also provide the implementation framework for simplifying the QoS-aware USN application programming.

Nr.	Design Question	Alternatives				
14	How is information exchange with a backend system organized?	exchange only on data level		exchange on application level		
15	What kinds of mechanisms are used for integration?	proxy server	service oriented	"sensor web" (OGC)	others	
16	Which application domains are supported with special services?	logistics applications	security applications	structural health mon.	other domains	
17	How is runtime support realized on the sensor node?	virtual machine		other solutions		
18	Which QoS parameters are used for network infrastructure?	message delay	loss	through-put	latency	others
19	Which QoS parameters are used for application support?	data accuracy	aggreg. delay	system coverage	system lifetime	others

Figure 3: Morphologic box for USN middleware

The morphologic box presented above does **not claim to be comprehensive or overarching**. During the coming standardization process it **has to be refined** and further developed.
Nevertheless the literature shows that the existing USN middleware systems feature totally different combinations of solution alternatives. Standardization has to cope with that fact.

1.4 A functional Model for Middleware

The reference model presented in the last chapter is by far too complex in order to be used for identification of the more important standardization issues. Therefore a simple functional model has been developed by SGSN based on the previous paperwork. It is shown in the following figure.

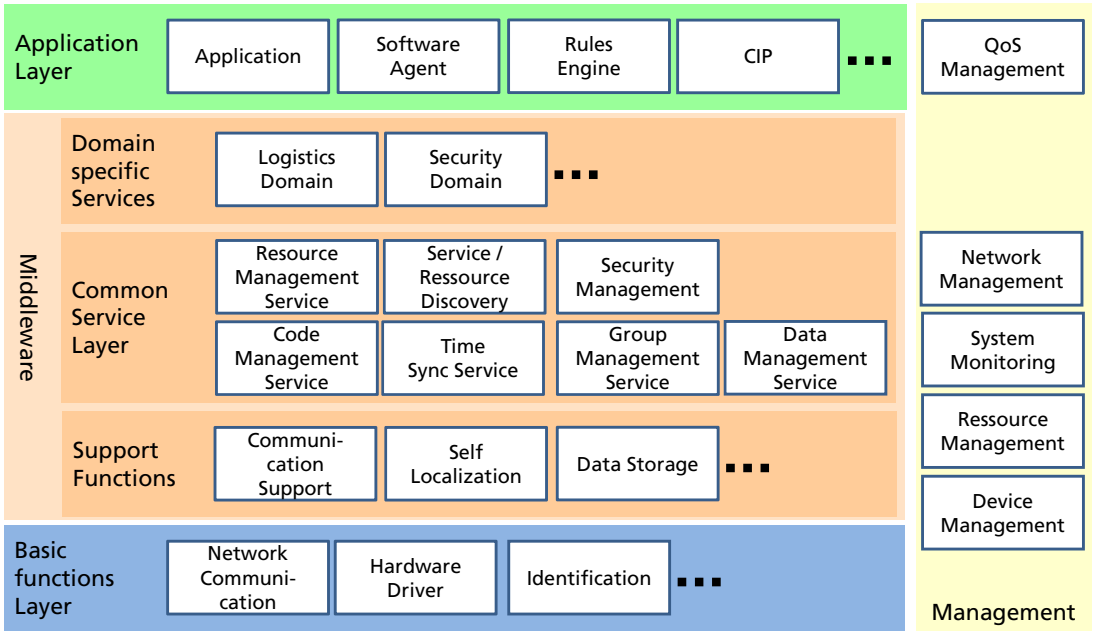

Figure 4: Functional model of USN Middleware

The middleware can be divided into three functional sub-layers: 1) Support functions are generic and locally available on the node. 2) The common service layer includes services which are available within the network and can be used by other nodes or via the gateway by the backbone network. The services offer basic functionalities which are common for sensor networks. 3) Domain specific services support the development of applications for a specific market segment or application area.

1.5 Analysis of Standardization Areas

For standardization the following working packages have to be addressed:

- Identification of the different functions or services on the three middleware levels (domain specific services, common services, support functions)
- Description of the functions on a first level (details have to be worked out by companies and research organizations working on middleware)
- Definition of interfaces for each and every module which is part of the functional model presented in figure 4

The functional model should be applied to middleware on the node, the gateway and the "middleware" or integration platform which is used in order to connect sensor network infrastructures to the information backbone within a company providing a sensor network based service to a customer.

The following aspects have to be addressed during standardization:

- Function interfaces on the node
- Function interfaces on the gateways

Development of implementation standards (connection between companies' IT systems and USN middleware) should be left to other standardization organizations (e.g. GS1).