Predefined Classification for Mixed Mode Environments

Arthur Herzog[1], Alejandro Buchmann
Technische Universität Darmstadt, Germany
{aherzog,buchmann}@dvs.tu-darmstadt.de

Abstract

In this paper we describe the classification we developed for Agent-based Middleware for Mixed Mode Environments (A3ME). In these environments we have some specific requirements and constrains so most existing ontologies are not appropriate. Therefore we first specify the requirements and constraints for the classification and in the second part we introduce the new classification.

1 Introduction

Mixed Mode Environments (MME) [1] are environments with different dimensions of heterogeneity: heterogeneous devices, heterogeneous software, and heterogeneous communication technologies. In MME we have many different devices and there are continuously new devices developed and introduced. To deal with these different kinds of devices we as humans intuitively classify them into different types. So that when ever we describe a specific device to somebody who does not know it yet, we first tell what type of device it is. For example when we want to describe an i900 we first tell that it is a smartphone.

Similar kind of classification we need for machine-to-machine and machine-to-human communication. When humans do this classification, not everyone will do it in the exact same manner. Actually it's even quite unlikely to have two exactly matching classifications defined by two different persons. Still it is usually not a problem for humans to understand each other. For machines it is different: here they have either to use the same underlying classification or the different classifications used by interacting machines have to be mapped to each other. This matching is not always possible and can require a lot of computation power. Therefore it would be reasonable to have a simple underlying classification to which most more complex classifications could be mapped.

[1] Supported by the DFG Research Group 1362, Cooperative, Adaptive and Responsive Monitoring in Mixed Mode Environments.

In Agent-based Middleware for Mixed Mode Environments (A3ME) [1] we pointed out a predefined lightweight ontology, which has to be used for classification of different devices, capabilities and services.

Our initial purpose was to find an existing classification that fits our needs. To check whether those are appropriate, we had to define the requirements (section 2). Afterwards we checked some existing ontologies with our requirements (section 3). In section 4 we describe the definition of the A3ME ontology and its usage.

2 Requirements

Here we have to define what must be classified and what the constrains for the classification are.

2.1 Classification Strategy

For our classification we decided for the *top-down strategy*. This way the classification will be more general as if we would drive it bottom-up by specialized use cases. The different kinds of devices must be classified to allow mapping of specific devices to a general class of devices with common characteristics. For example, all kinds of cellphones, PDAs and smartphones could be classified as mobile phones. In the second step, the different capabilities have to be classified into groups of related capabilities, e.g. sensing capabilities, actuator capabilities, etc. These can than be further sub classified into more specific subtypes.

Since MME are intrinsically heterogeneous and dynamically changing, it is obvious that the classification will not be complete. This means there always will be devices and capabilities that are not covered by this classification. Therefore *extensibility* of the classification is essential. Never the less for any extension the basic classification still has to stay valid.

2.2 Use of the Classification

In MME you cannot rely on a reliable connection and availability of any server. Therefore here a decentralized approach has to be used. Consequently the classification must be available at the devices themselves, at least the subset of the classification that is relevant for the device. Furthermore the classification must be usable on resource-constrained devices like TelosB sensor motes [6].

Concerning the constrained communication in terms of bandwidth and energy usage, the overall *size* of the classification should be kept small. Our goal is to encode any classification item in one byte.

Because of the limited computation and storage capabilities of some devices in MME, the classification *complexity* in terms of its depth should also be kept low.

3 Related work

During the search for an appropriate classification/ontology we looked at present existing ontologies. In the following we show some of them and explain, why they were not appropriate for our task.

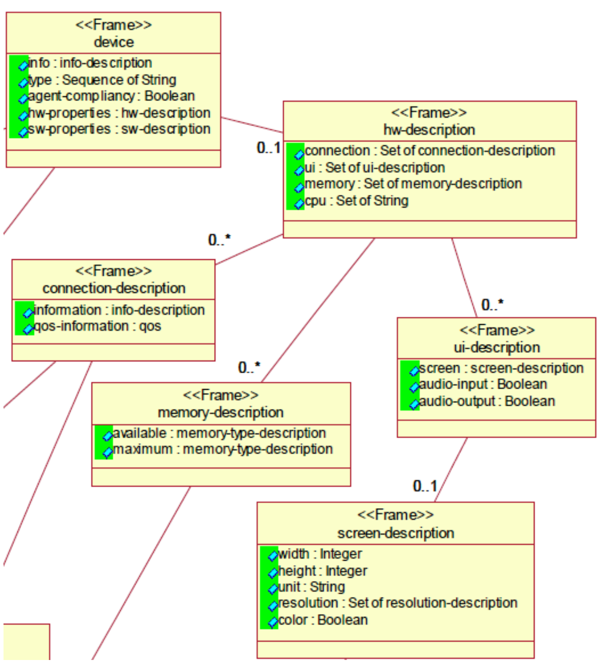

Figure 1: Excerpt of the FIPA device ontology

3.1 FIPA Device Ontology Specification

FIPA Device Ontology [5] allows describing software and hardware of a device. For hardware description it uses three subtypes: connection, memory and user interface (Figure 1). It does not define a hierarchy of capabilities for devices, what we want to use the ontology for.

3.2 OntoSensor Ontology

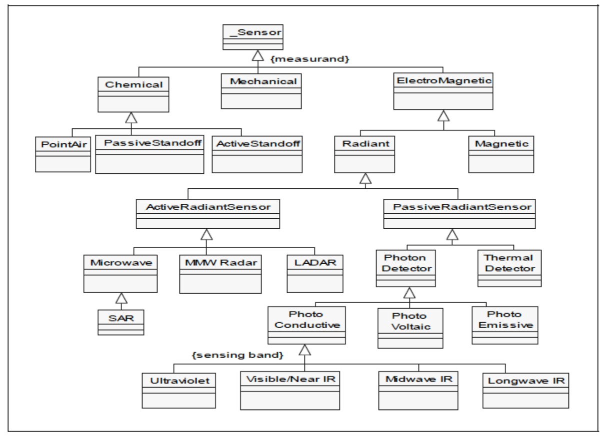

Figure 2: Excerpt of sensor hierarchy within OntoSensor

OntoSensor [2], [3] is a prototype sensor knowledge repository, which includes definitions of concepts and properties adopted (in part) from SensorML [7], extensions to IEEE SUMO [8] and references to ISO 19115 [9]. Figure 2 shows an excerpt from OntoSensor ontology. This ontology has a very detailed classification. The classification is done dependent from the way it is measured.

For our purpose this classification is too detailed and too complex. For example, to describe a simple light sensor you have to go through six classification levels.

3.3 SOPRANO Context Ontology

Soprano is an ontology from the ambient living area, therefore it is quite specific and limited to the task. With respect to device capabilities it only distinguishes between sensors and actuators (Figure 3).

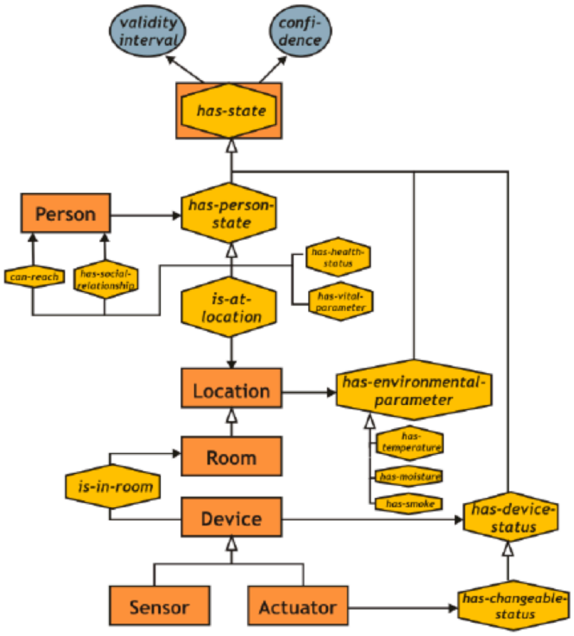

Figure 3: Soprano context ontology

4 Classification Definition

First level of the classification deals with different aspects needed to be classified in MME. Those are IDs, devices, capabilities, services, data, properties and other (Figure 4). Some of these are further sub classified. Figure 6 shows the complete classification. In Appendix A the static encoding of the classification is listed. The classification is also available as an OWL file[2].

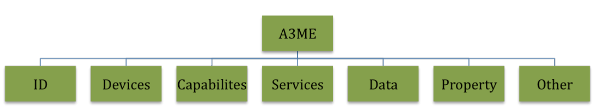

Figure 4: First level A3ME classification

4.1 IDs

In MME there can be different networks, each using its own addressing scheme to identify the participating nodes. Therefore, we added also a simple classification for the ID. ID is further sub classified into local and global IDs.

4.2 Devices

For devices we first identified seven different classes of devices: tag, mote, mobile, workstation, server, vehicle and multimedia.

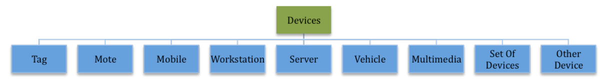

Figure 5: Classification of devices

The Tag class stands for passive devices like RFID tags. The Mote class devices describe small resource constrained devices like those used in wireless sensor networks. Mobile class contains all kind of cell phones, PDA and similar devices. Workstation class can be used to describe all kinds of personal computers. Server devices are special computers capable of storing and/or computing large amount of information. Vehicle class shall be used to describe all kind of vehicles like remote controlled cars, autonomous robots, spacecrafts, etc. Multimedia covers devices like TV, HIFI, radio, etc.

An additional device class can be used to describe sets of devices. To extend this device classification class "other device" can be used.

4.3 Capabilities

Capabilities here are functional properties, which describe some abilities of the device. For capabilities we first define top-level capability classes: sensor, actuator, human interface devices (HID), energy, communication, CPU, storage and other capability. These class types shall roughly describe the kind of capability. Sensor, actor, HID, energy and storage classes are further sub classified. Communication and CPU classes are not further sub classified, because here it is more appropriate to use parameters to describe specific capabilities of these classes. For example, communication capability can be described by filling the following parameters: communication media, communication standard, protocol, bandwidth and so on. A CPU can be described with following parameters: frequency, number of cores and type of instruction set used, etc.

4.4 Services

For services we identified three basic groups: hardware related, software related and real world services. Hardware related services enable access to hardware capabilities of a node, like accessing sensors or actuators. Software services are all services, which are not directly dependent on some hardware capabilities. Typical software services are computing functions, virtual machines for “mobile software agents”, etc. The third group of services covers services in the real world, like food delivery, transportation, etc.

[2] http://www.dvs.tu-darmstadt.de/staff/aherzog/a3me/a3me.owl

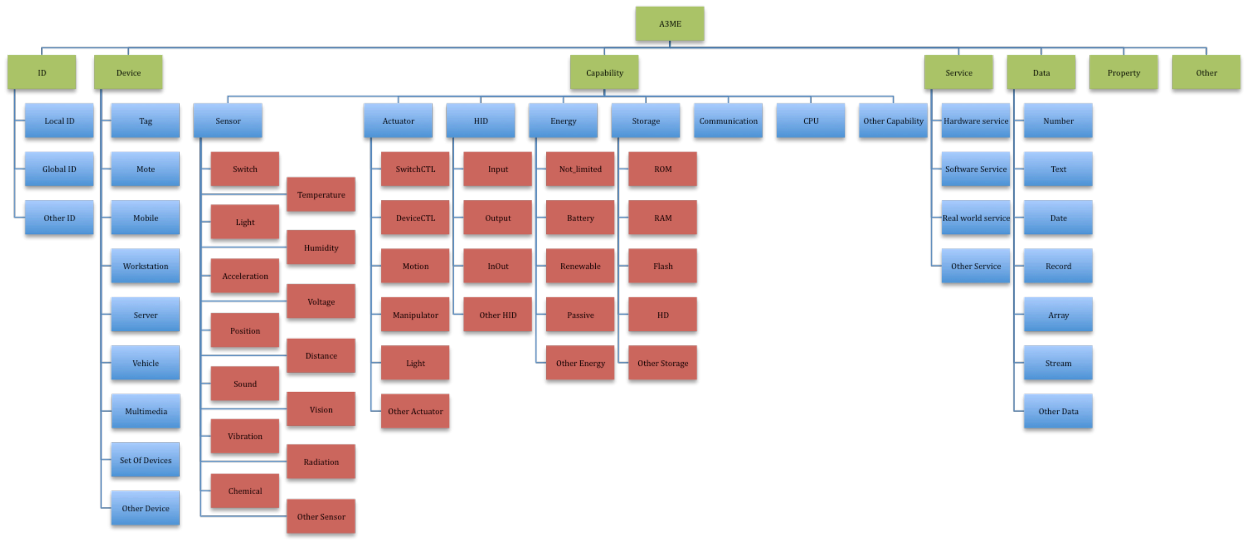

Figure 6: A3ME classification

4.5 Data

We decided to add a basic classification of data types, to make this basic classification complete and to allow specifying, for example, the type of data someone is requesting.

4.6 Properties

The properties branch describes characteristics that are not capabilities of the device. These additional characteristics are often used for self-description and discovery. For example, the manufacturer and the owner of a device are typical properties.

4.7 Other

Each branch in this classification has an element “other …”. These elements are thought as points at which the classification can be extended. If, for example, the element “other device” is extended, it would mean, that all existing device types in the classification are not appropriate to describe the device being classified.

5 Conclusion

In this paper we described a new simple (on purpose) but extendable classification, which can be used to classify devices, capabilities, services, IDs, properties and data in a MME. The encoded classification can be used for initial self-description and discovery independent of the underlying communication technique.

6 References

[1] A. Herzog, D. Jacobi, and A. Buchmann. A3ME - An Agent-based Middleware Approach for Mixed Mode Environments. In The Second International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2008), pages 191 – –196, Valencia, Spain, oct 2008.

[2] D.J. Russomanno, C. Kothari and O. Thomas “Building a Sensor Ontology: A Practical Approach Leveraging ISO and OGC Models”, *The 2005 International Conference on Artificial Intelligence*, Las Vegas, NV, 2005, pp. 637-643.

[3] OntoSensor project, University of Memphis http://www.memphis.edu/eece/cas/projects.php, 2009.

[4] Open Geospatial Consortium, Inc., OpenGIS Sensor Model Language (SensorML), 2000.

[5] Foundation for Intelligent Physical Agents. FIPA Device Ontology Specification, 2002.

[6] Crossbow Technology Inc. Telosb datasheet. http://www.xbow.com, 2006.

[7] SensorML Implementation Specification, OGC-05-086r2, Mike Botts; Open Geospatial Consortium Inc., University of Alabama in Huntsville, 2006.

[8] I. Niles, A. Pease “Towards a standard upper ontology”, Proceedings of the international conference on Formal Ontology in Information Systems, Maine, USA, 2001.

[9] ISO 19115 Geographic Information - Metadata. International Organization for Standardization (ISO), Geneva, 2003.

Appendix A: Classification List with Numeric Encodings

	Level 1	Level 2	Level 3	Code
A3ME				0
	ID			1
		Local ID		2
		Global ID		3
		Other ID		4
	Device			5
		Tag		6
		Mote		7
		Mobile		8
		Workstation		9
		Server		10
		Vehicle		11
		Multimedia		12
		Set Of Devices		13
		Other Device		14
	Capability			15
		Sensor		16
			Switch	17
			Temperature	18
			Light	19
			Humidity	20
			Acceleration	21
			Voltage	22
			Position	23
			Range	24
			Sound	25
			Vision	26
			Vibration	27
			Radiation	28
			Chemical	29
			Other Sensor	30
		Actuator		31
			Switch controller	32
			Device controller	33
			Motion	34
			Manipulator	35
			Other Actuator	36
		HID		37
			Input	38
			Output	39
			InOut	40
			Other HID	41
		Energy		42
			Not limited	43
			Battery	44
			Renewable	45
			Passive	46
			Other Energy	47

	Level 1	Level 2	Level 3	Code
		Storage		48
			ROM	49
			RAM	50
			Flash	51
			HD	52
			Other Storage	53
		Communication		54
		CPU		55
		Other Capability		56
	Service			57
		Hardware Service		58
		Software Service		59
		Real world Service		60
		Other Service		61
	Data			62
		Number		63
		Text		64
		Date		65
		Record		66
		Array		67
		Stream		68
		Other Data		69
	Property			70
	Other			71