Interoperability and ontology for heterogeneous systems

MICHAL ŠÍR
PETR FIEDLER
VÁCLAV KACZMARCZYK
Control of instrumentation
Brno University of Technology
Kolejní 2906/4 612 00 Brno
CZECH REPUBLIC
xsirmi00@stud.feec.vutbr.cz, fiedlerp@feec.vutbr.cz, xkaczm00@stud.feec.vutbr.cz

Abstract: - This paper deals with generic principles that are used to achieve interoperability in heterogeneous systems. Interoperability classes are provided and operations for working with ontologies are defined. Based on these operations, rules and classes an interoperability framework intended for exchange of data in sensor networks and data acquisition systems is created.

Key-Words: - Interoperability, ontology, heterogeneous system, information flow, automation device

1 Introduction
Communication among heterogeneous systems has been plagued with problems since first communication protocols were created. Nowadays, when communication protocols directly surround us, it’s not possible to develop unified communication protocol, which would enable us to communicate with the majority of the equipment. Such a protocol cannot be realized due to both economic and technical challenges. The aim of this work is to create an interface with a certain degree of abstraction, that would enable to face the existing situation, i.e. to face the the interoperability issues encountered when interconnecting heterogeneous systems. We believe that solution can be found by creating upper ontology for the information being exchanged. This upper ontology type will create the concept for the exchange of both process (sensor) data and device status data.

2 Interoperability
ISO / IEC 2382 Information Technology Vocabulary defines Interoperability as the ability to communicate, to execute programs or transfer data among different functional units. The interoperability problem can be seen in different views. Each device contains a representation formats (syntaxes), data organization (structure) and different points of view level (semantics) which are used in the defined device [1]. For easy interoperability achievement there are defined interoperability classes [2].

1. Class A Communication Interoperability: The devices are able to exchange data, in the same syntax, using a specific communication stack behavior (service interface, protocol, signaling and physical media) and network management scheme. Two devices are also able to exchange information with each other, with the same semantics, using a specific application process model, generic device behavior, variable dictionary, and datastore architecture.

2. Class B Application Interoperability: The devices are able to coordinate control, with the same device-specific behavior, using a specific control strategy (i.e., cyclic, timed, phase and adaptive).

3. Class C Tool system Interchangeability: The devices are able to support the tool system’s required levels of response, capacity, fault tolerance and reliability, by using a common device architecture, compatible physical packaging, and resource management scheme.

3 Ontology

The term "ontology" can be defined as an explicit specification of conceptualization. Ontologies capture the structure of the domain, i.e. conceptualization. The conceptualization describes knowledge about the domain, not about the particular state of affairs in the domain [3]. In this context ontology is used for the creating of ontological commitments. The ontological commitment is an agreement of using the dictionary (i.e., asking questions and making assertions) in the manner that is consistent with respect to the specific ontology theory.

3.1 Operations with Ontologies
With respect to the complexity and degree of abstraction it’s necessary to create several independent ontologies, which will be interdependent. First, the upper
ontology has to exist as it will serve as a communication bridge to other lower ontologies. Due to a modular architecture we need ontological operations, allowing “cooperation” among ontologies. Basic operations as defined by Marek Obitko [3] are listed below:

- **Merge of ontologies** means creation of a new ontology by linking up the existing ones.
- **Mapping** from one ontology to another one is expressing of the way how to translate statements from ontology to the other one.
- **Alignment** is process of mapping between ontologies in both directions whereas it is possible to modify original ontologies so, that suitable translation exists (i.e., without losing information during mapping).
- **Refinement** is mapping from ontology A to another ontology B so, that every concept of ontology A has equivalent in ontology B.
- **Unification** is aligning all of the concepts and relations in ontologies so, that inference in one ontology can be mapped to inference in other ontology and vice versa.
- **Integration** is a process of looking for the same parts of two different ontologies A and B while is developing new ontology C. That allows translation between ontologies A and B and so allows interoperability between two systems where one use ontology A and the other use ontology B. The new ontology C can replace ontologies A and B or can be used as an interlingua for translation between these two ontologies.
- **Inheritance** means that ontology A inherits everything from ontology B. It inherits all concepts, relations and restrictions or axioms.

### 3.2 Ontology description

For a correct description of the ontology used OWL language. OWL (Ontology Web Language) is markup language developed by W3C organization for creating ontologies used in semantic web environments. Language itself consists of three parts shown in the Figure 1. For our case the most appropriate is to use the OWL DL language.

The OWL DL is based on a descriptive logic. Creating ontologies, determining the ontological hierarchy and checking consistency is possible with this language. Using OWL DL it can be assured that all conclusions will certainly be computable and all calculations will be finished in a finite time; this property has been described in [4].

### 4 Device concept

In the field of automation the process data exchange in general is based on real and virtual devices that provide inputs and outputs. The devices are sometimes described by device profile, that restrict device interfaces. However devices in general (in the context of process data exchange), are consisted from:

- **Sets of Input/Output values**: Set of all process data input/output values and associated attributes that describe the I/O value and its quality;
- **Sets of other HW modules**: Sets of all modules that make up the device;
- **Sets of SW modules**: Sets of all software modules that make up the firmware of the device;

the input and output channels have to have the same attributes to enable trouble-free sharing of information. It requires equivalent structures at the sender and at the receiver to prevent distortion of information. Proposed
concept of an input/output channels, shown in figure 3, consist from:

- **Engineering Value**: Quantification of the input/output information.
- **Validity**: Quantification of validity of information stored in the Engineering Value; the validity is gained by evaluation of the status of the provider of the engineering value; this information quantifies truthfulness of the provided information according to the best knowledge of the information provider. Such quantification of truthfulness “upgrades” the information exchange to a form of knowledge sharing.
- **Timestamp**: Timestamp captures time of the measurement; system wide clock synchronization (global system clock) is required for meaningful timestamping.

![Fig. 3. Concept of an input/output channel](image)

The validity describes to what extent the information provider (sender) believes, that the provided information represents the “truth”. It is a crucial element in the concept because it extends the raw information to some form of knowledge. In sensor fusion applications it enables to assign weights to various sources of information.

Important part of any Device is a Device status. It’s a concept of capturing and enumerating of actual state of the device HW and SW subsystems with respect to process data exchange; device status should cover status of all functional parts of the device (e.g. power supplies, communication interfaces, clocks, IO modules, etc.) Device status thus represents device ‘health’.

The status for each subsystem has to be defined as a concept. The validity concept has to include probability of a failure, estimated degradation due to use/abuse and semantic distance between the detected or estimated state and an ideal state. The ability to evaluate the status of the value provider requires extensive self-diagnostic features of the automation devices and their subsystems. It is necessary to elaborate how the device subsystems should diagnose themselves (evaluate their own health) and how to report their estimated ‘health’ in a uniform way. Some inspiration can be found with S.M.A.R.T technology used in hard disk drives.

The validity concept we propose consist of:

- A concept that defines both semantic distances between various possible states and a method for evaluation of the semantic distance between the detected state and an ideal state;
- Probability of failure that would prevent correct evaluation of semantic distance from the ideal state of the device; (probability of an undetected or a wrongly classified failure).

Such validity concept is important, because it enables to express fact, that gained information or knowledge is not necessary truth. The possibility to evaluate a complex status information into a single numeric value or simple unified structure will help to assure that if communicating systems understand the concept of validity and if some information is lost and/or faked, then the recipient is informed that the gained information is in a quantified extent distorted or suspicious.

### 4.1 Engineering value

Engineering value in our vision is not a simple value. Again it is a concept that enables to describe the physical quantity according to the best practices in the field of measurement.

![Fig. 4. Concept of an engineering value](image)

In this concept, shown in figure 4, the input/output engineering value consists following attributes:

- **Value**: Numerical quantification of the input/output.
- **Engineering unit**: Qualitative description of engineering unit used for quantification of the value; from the methods used for description of engineering units known to the authors a principle defined by Bruce Hamilton [5] and adopted in the IEEE 1451 standard is the most universal and the least limiting one.
- **Uncertainty**: The uncertainty of the measurement as defined in the ISO "Guide to the Expression of Uncertainty in Measurements"; this information provides a range, which (as the information producer believes) with defined probability contains true value. This information quantifies random measurement error with an assumption that the unit is not faulty (no systematic error due to device fault or incorrect calibration).
The Engineering value together with Validity and Timestamp represent a knowledge of state of an input/output in given time and it's consisted from:
1. Data (value).
2. Attributes defining semantics of data (engineering units).
3. Quantification of truthfulness and/or validity of the provided data (Uncertainty).

5 Conclusion
Three classes of interoperability rules are presented in this paper. Operations with ontologies and selection of the appropriate language are provided. For the future work the upper ontology for heterogeneous systems with the certain level of abstraction needs to be designed. We believe that such ontology for sensor networks and other automation systems needs to be elaborated above the concept of a generic automation device utilizing the before mentioned interoperability classes. Lower ontologies for specific devices should be designed based on the upper ontology using operations from section 3.2. As a result of these operations a communication “bridge” through upper ontology connecting specific devices and systems will be enabled.

6 Acknowledgments
This work has been supported in part by Ministry of Education, Youth and Sports of the Czech Republic (Research Intent MSM0021630529 Intelligent systems in automation (VZ UAMT), Grant Agency of the Czech Republic (102/09/H081 SYNERGY – Mobile Sensoric Systems and Network) and by Brno University of Technology. Without kind support of the above-mentioned agencies and institutions the presented research and development would not be possible.

References: