

An ontology-based method to merge and map management information models*

Jorge E. López de Vergara, Víctor A. Villagrà, and Julio Berrocal

Departamento de Ingeniería de Sistemas Telemáticos,
Universidad Politécnica de Madrid (DIT-UPM).

E.T.S.I. de Telecomunicación, Ciudad Universitaria, 28040 Madrid, Spain.

{jlopez, villagra, berrocal}@dit.upm.es

Abstract:

The multiplicity of network management models (SNMP, CMIP, DMI, WBEM...) may imply in some scenarios the use of multiple management information models defining the resources to be managed. In these scenarios it is necessary that a manager can have a common view of all resources to monitor and control them in an integrated way. However, information models cannot be easily integrated due to the difficulties that arise when dealing with the semantics they contain. Syntactic translations also known as recasts are commonly used, but they only generate set of unlinked models with the same format instead of a common view of the managed resources.

The integration of partially overlapped information models has also been studied in the ontology field. Ontologies exactly focus on the meaning of those concepts composing an information model, and knowledge systems using ontologies also need such integration to interoperate. This paper will show the application of ontology-based techniques to improve the integration of current network management models, explaining the possibilities to reach a semantic translation.

Keywords: Ontology, Network Management, Information Models Integration, Semantic Translation, Merge and Map.

1 Introduction

Currently, several Integrated Network Management models coexist with different management information models defining the resources to be managed. Different approaches have been applied to translate specifications and interactions among OSI-MS (Open Systems Interconnection, Management Systems), SNMP (Simple Network Management Protocol), CORBA (Common Object Request Broker Architecture) or DMI (Desktop Management Interface). For instance, IIMC (ISO Internet Management Coexistence) or JIDM (Joint Inter-Domain Management) are some results of these efforts. The final objective is that a manager can have a common view of all resources to monitor and control them in an integrated way.

However, current methods only provide recast translations that work in a syntactic way: a mapping between the information languages is done instead of a mapping between the specifications. Although this information is translated using the same format, the managed resources specifications are still a set of unlinked models. The information remains isolated because its semantics includes the relationships that connect it to the rest of the model.

In prior works [1, 2, 3], the use of ontologies has been proposed to deal with the semantics of the management information: they can improve the expressiveness of the management languages, adding formal axioms and constraints to the specifications. Moreover, ontology techniques for merging and mapping specifications can also be applied to information management models. Taking these proposals as a starting point, this paper details these methods to combine and match ontologies and their adaptation for the integration of

* This work has been partially funded by the Spanish Ministry of Science and Technology under the project GESEMAN (TIC2002-00934).

management information models, improving current approaches to reach a semantic translation. In this way, as shown in Figure 1, a common specification of all managed resources can be obtained by merging different models. On the other hand, it is also necessary to translate the merged model the manager uses to those models defined for every agent. For this, a mapping ontology is specified that defines a set of rules to semantically translate the management information.

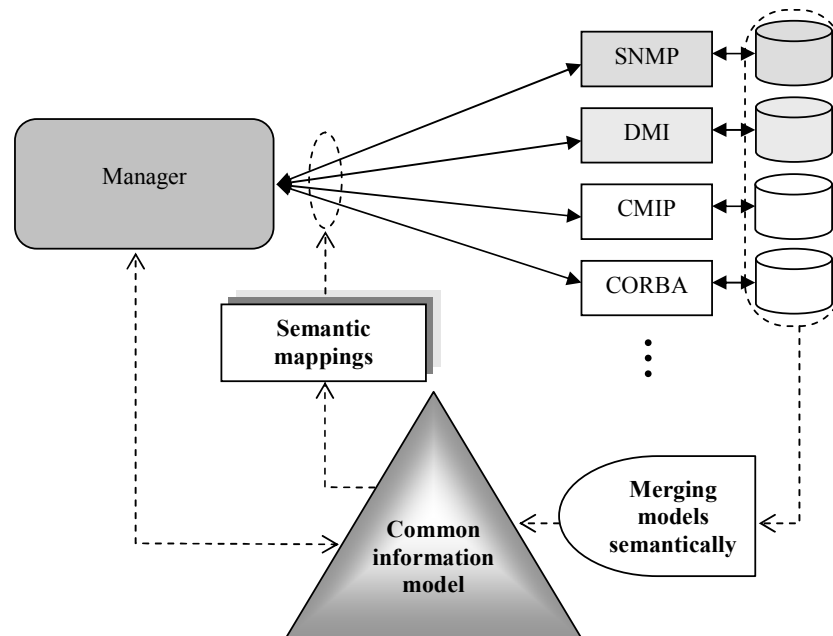


Figure 1. Paper objectives

This paper is structured as follows. First of all, ontologies are presented. Then, an ontology-based method to merge and map management information is explained. Next, a case study is shown that applies this method. Some conclusions are finally given.

2 Ontologies

Ontologies, mainly used in the Artificial Intelligence field, are explicit and formal specifications of shared conceptualizations [4]. They have been relevant for the development of intelligent agents and currently are a key technology of the Semantic Web. Some applications to network management have also been presented recently. A mapping between SMIng (Structure of Management Information, next generation) and RDF (Resource Description Framework) has been proposed in [5] so that intelligent agents can use SNMP MIBs as their common vocabulary. CIM (Common Information Model) metaschema has also been mapped into OKBC with a similar purpose in [6]. This paper focuses on other application of ontologies: the integration of network management information taking into account the semantics included on it.

Solutions applied to integrate different ontologies deal both with the syntax of the description languages and the meaning of the described information, addressing this problem also from the semantics viewpoint. Approaches in this scope include some methods that merge, align or map the information:

- When merging and aligning ontologies, a new ontology is generated that has all the concepts included in the initial ones. For this, different proposals based on heuristics exist: candidate concepts or properties to be merged can be found if they are named with a similar string [7, 8] or if the values of their instances are also similar [9].

- When mapping ontologies, several rules are defined that allow a semantic translation of their instances. For this, a mapping ontology [10, 11] can be defined that includes all the concepts of these rules, such as the elements to be mapped, or the formula used for the translation.

Next section takes advantage of these methods and adapts them for the semantic integration of management information specifications.

3 M&M: a method to integrate management information models semantically

Even if all management information models are translated into the same language, the semantics they contain is not integrated. It is necessary to carry out other procedures that identify their meaning. This can be done by leveraging the merging and mapping techniques used in the ontology field. This section presents a new integration method, based on the combination of different specifications in a common model, defining the mapping rules with the initial models, as shown in Figure 2.

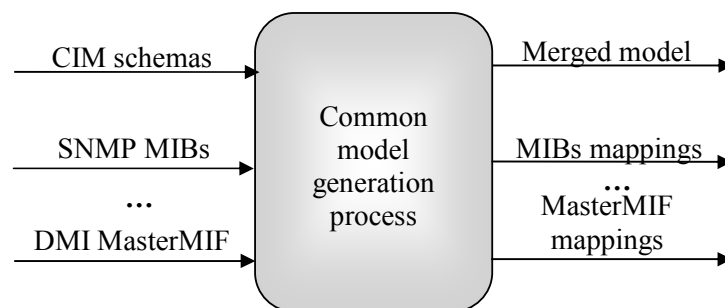


Figure 2. Process of merging and mapping management information.

To assist in this process, a method called M&M (Merge and Map) has been developed. It proposes a set of steps to help in the procurement of both the common model and the mapping rules. This differs from usual ontology integration methods, which are about merging or mapping but not both processes in parallel.

The merging part is based on the process explained in [7] adapted for the network management particularities. Other techniques are not applicable to network management information, because they only deal with classes [8] (properties are not taken into account) or instance values [9] (which are not known when merging management information models).

The aim of this method is to help the person carrying out this process. For this, the following set of heuristics is applied to find candidates to be merged with high probability:

- Candidates by similar character strings. Two classes or properties are candidates to be merged with high probability if they have a similar substring included in their identifiers or in their descriptions. Synonyms can also be used if they are available.
- Candidates by similar inheritance hierarchy. Two classes are candidates to be merged with high probability if their parent classes are similar, because child classes of a class are usually similar to the child classes of another class that has been merged with the first one.
- Candidates by property domain. Two properties are candidates to be merged with high probability if the classes containing them are also similar. At the same time, two classes are candidates to be merged with high probability if the properties they contain are also similar.

To describe the mapping rules that translate the information instances from a concrete model to the common model a simple mapping ontology has been defined, using DAML+OIL (DARPA Agent Markup Language plus Ontology Inference Layer) [12] as the specification language. Figure 3 shows with a class diagram a representation of this ontology. It is based on ideas contained in [11], but reducing its complexity. A simpler approach [10] is not applicable because it only deals with classes.

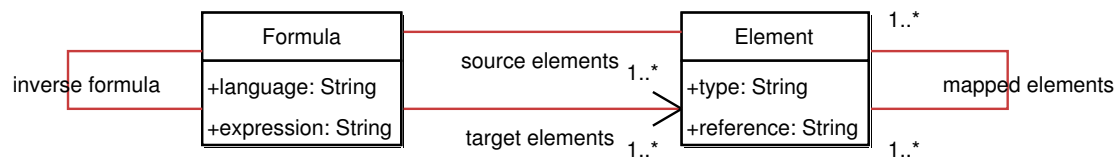


Figure 3. Mapping ontology

This ontology has two concepts: each `Element` composing an ontology (classes, properties, etc.) has a translation `Formula`. Every `Element` has some properties such as the `type` or the `reference` to its specification. Every `Formula` has an `expression` written in a concrete language to translate the set of `source` and `target` elements. Relationships among `mapped` elements and `inverse formula` are also included. With this ontology any mapping of those identified before can be addressed. Other approaches, such as the `MappingStrings` qualifier used in CIM, can only represent direct mappings.

Typical mappings among management information models have been identified. They can be the following ones or a combination of them:

- Direct, if it is a 1:1 relationship in which no transformation is needed. In this case, the value contained in an element in both models is exactly the same. Direct mappings are most common ones and are proposed by default.
- Value set, if it is a 1:1 relationship in which for each value of an element there is another value for the element of the other model. This kind of mapping is proposed if at least one of two merged properties has an enumeration of possible values.
- Data types, if it is a 1:1 relationship in which the elements of both models have different data types that have to be converted.
- Arithmetic operation on one element, if it is a 1:1 relationship in which the value of one element is obtained calculating it with the value of the element of the other model. This kind of mapping is useful if the measurement units of both elements are different.
- Arithmetic operations on some elements, if it is a 1:n relationship in which the value of one element is obtained by arithmetic combination of the values of some elements of the other domain. In this case, the user should define the mapping expression.
- Character strings, if it is a 1:n relationship in which the value of one element is composed by concatenating different strings, which are the values of the elements in the other domain. Once again, the user has to define the mapping rule.

Figure 4 shows the activity diagram that describes the method M&M. Gray activities are those performed by a user and those in white are automatically completed by the system. In brief, this method consists on the identification of similar classes and properties, using commented heuristics, to merge them. At the same time, every element of the mapping ontology is automatically defined at the beginning, and later, associated formulas are created when merging the elements. The person in charge of this process must validate every operation proposed by the method, and can also define different ones. The final result is a

common model and a set of instances of the mapping ontology representing the rules to translate the merged models, as stated before.

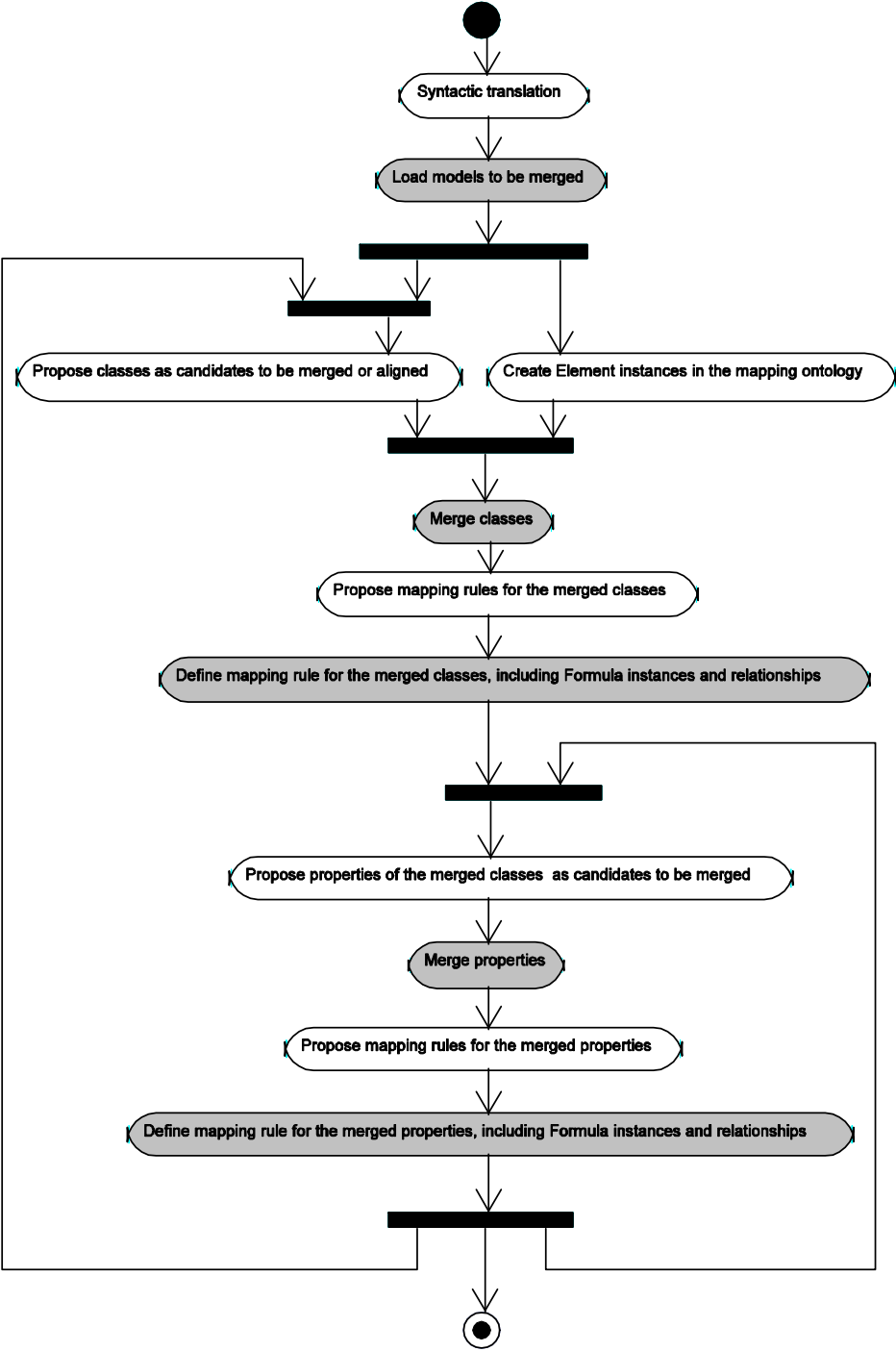


Figure 4. M&M activity diagram

A manager based on this proposal would work in the following way. If it needs to obtain all instances of a concrete element of the common model, it will look for it in the mapping ontology, finding at the same time the formula and mapped elements of the merged models. It will access each management domain containing this information to get it. Applying the expression contained in the formula it will translate the value of the instances obtained in each domain to the common model.

4 Using M&M: integrating the HOST-RESOURCES-MIB in the CIM schema

To validate the proposed merging and mapping method, it has been applied to a case study in which the HOST-RESOURCES-MIB has been integrated with the CIM schema. Following the M&M activity diagram, the first task has been the translation of both specifications to a common language, which could be MOF (Managed Object Format). In this case, as proposed in [13], tables and groups of objects of the MIB have been considered as classes, and external indexes as inheritance relationships. These information specifications are shown as class diagrams in Figure 5 and Figure 6.

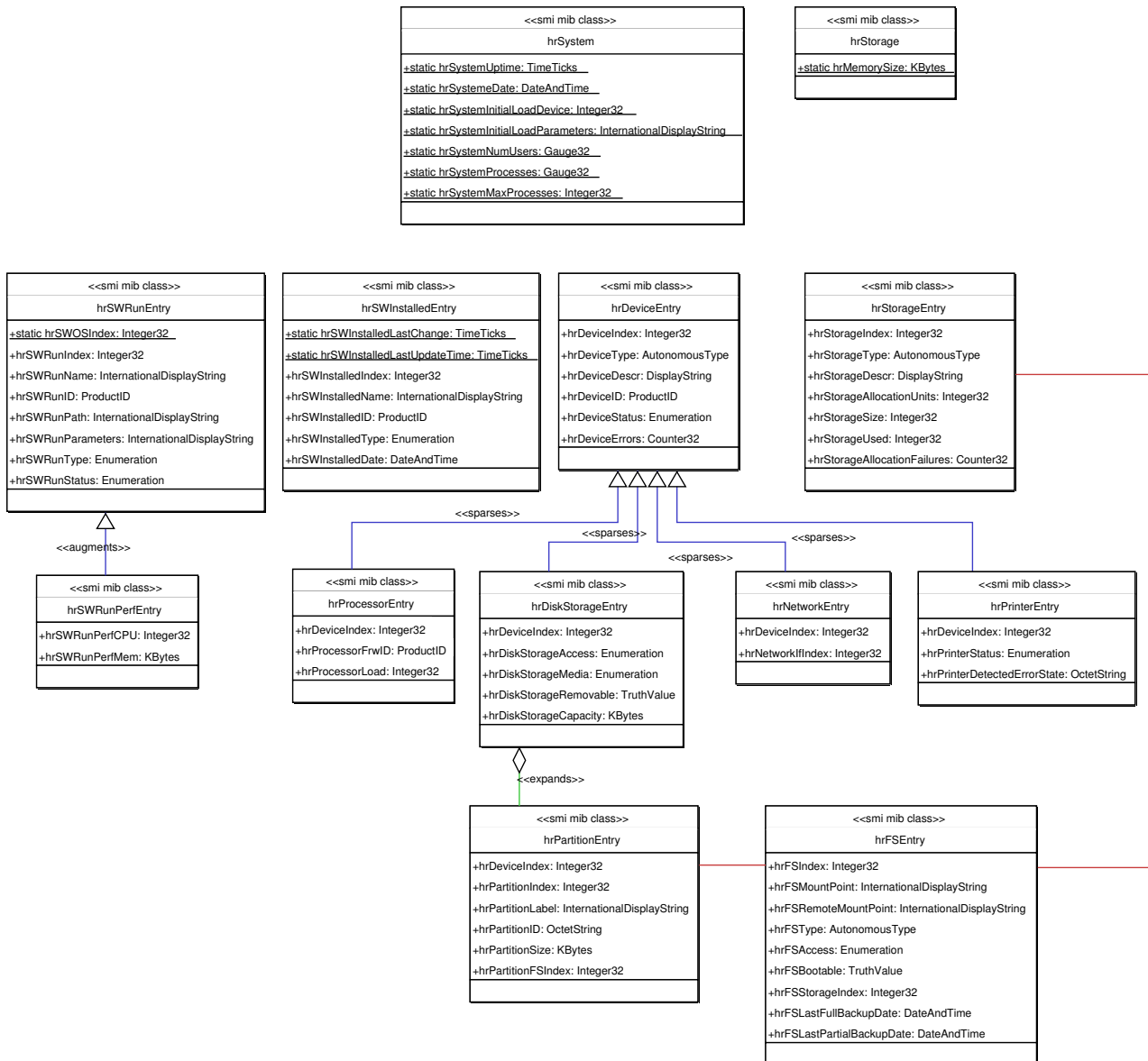


Figure 5. HOST-RESOURCES-MIB groups and tables represented as classes

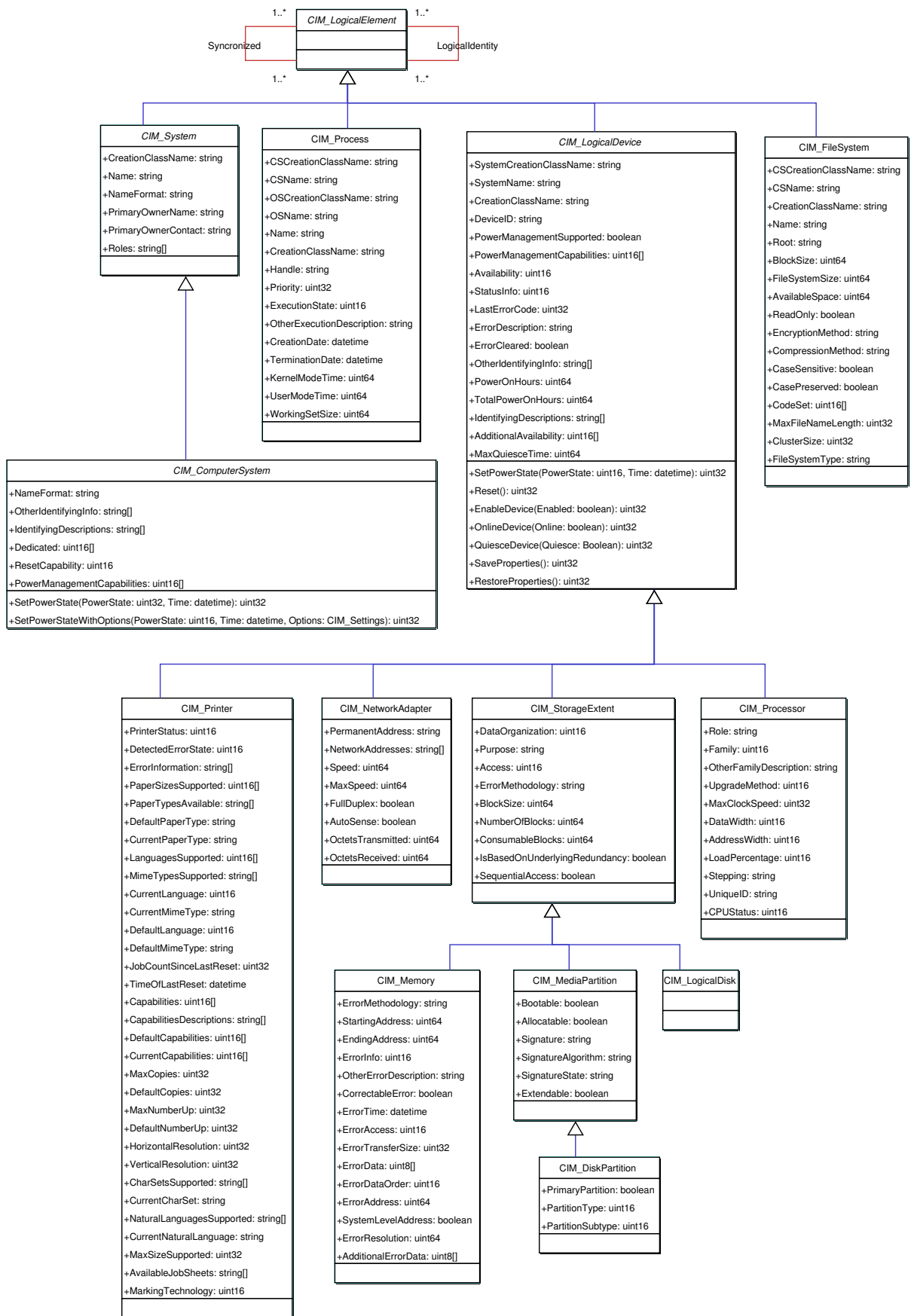


Figure 6. CIM schema subset

Next, using proposed heuristics, some table entries and classes, including their columns and properties respectively, have been taken as candidates to merge them:

- Candidates by similar character strings:
 - hrSystem with CIM_System and CIM_ComputerSystem, because they share the System substring.
 - hrDeviceEntry with CIM_Device, because they share the Device substring.
 - hrProcessorEntry with CIM_Processor, because they share the Processor substring.
 - hrDiskStorageEntry, hrStorage and hrStorageEntry, with CIM_StorageExtent, because they share the Storage substring.
 - hrPartitionEntry with CIM_MediaPartition y CIM_DiskPartition, because they share the Partition substring.
 - hrNetworkEntry with CIM_NetworkAdapter, because they share the Network substring.
 - hrPrinterEntry with CIM_Printer, because they share the Printer substring.
 - hrSWRunEntry, hrSWRunPerfEntry, hrSWInstalledEntry, and hrFSEntry do not have any string similarities with the CIM schema, unless synonyms of SW (Software), Run (Process) or FS (File System) are defined.
- Candidates by similar inheritance hierarchy:
 - hrDeviceEntry child classes with those of CIM_Device.
 - hrProcessorEntry with CIM_Processor.
 - hrDiskStorageEntry with CIM_StorageExtent.
 - hrNetworkEntry with CIM_NetworkAdapter.
 - hrPrinterEntry with CIM_Printer.
- Candidates by property domain, joint with similar character strings:
 - hrDeviceEntry and CIM_Device properties.
 - hrDeviceID with DeviceID.
 - hrDeviceStatus with StatusInfo.
 - hrDeviceErrors with LastErrorCode, ErrorDescription and ErrorCleared.
 - hrProcessorEntry and CIM_Processor properties.
 - hrProcessorFwdID with UniqueID.
 - hrProcessorLoad with LoadPercentage.
 - hrDiskStorageEntry, hrStorage, hrStorageEntry and CIM_StorageExtent properties.
 - hrDiskStorageAccess with Access.
 - hrPrinterEntry and CIM_Printer properties.
 - hrPrinterStatus with PrinterStatus.
 - hrPrinterDetectedErrorState with DetectedErrorState.

At the same time, different `Element` instances have been generated. Next lines show an example written in DAML+OIL with the definition of some classes and properties from both models.

```

<Element rdf:ID="CIM_Process"
  rdfs:label="CIM_Process"
  type="class"
  reference="DMTF/CIM/CIM_Process" />
<Element rdf:ID="KernelModeTime"
  rdfs:label="CIM_Process.KernelModeTime"
  type="property"
  reference="DMTF/CIM_Process.KernelModeTime" />
<Element rdf:ID="UserModeTime"
  rdfs:label="CIM_Process.UserModeTime"
  type="property"
  reference="DMTF/CIM_Process.UserModeTime" />

<Element rdf:ID="hrSWRunPerfEntry"
  rdfs:label="hrSWRunPerfEntry"
  type="class"
  reference="IETF/HOST-RESOURCES-MIB/hrSWRunPerfEntry" />
<Element rdf:ID="hrSWRunPerfCPU"
  rdfs:label="hrSWRunPerfEntry.hrSWRunPerfCPU"
  type="property"
  reference="
    "IETF/HOST-RESOURCES-MIB/hrSWRunPerfEntry.hrSWRunPerfCPU" />

```

Finally, when two or more elements are merged, a `Formula` for their translation is also proposed. Code included below shows an example of such a formula, in which an expression has been set up to obtain the value of `hrSWRunPerfCPU` with an arithmetic operation combining the values of `KernelModeTime` and `UserModeTime`. The expression is written in a fictitious language called `MapTrans`, similar to JavaScript.

```

<Formula rdf:id="KernelModeTime,UserModeTime->hrSWRunPerfCPU">
  <language>MapTrans</language>
  <sourceElements rdf:resource="#KernelModeTime" />
  <sourceElements rdf:resource="#UserModeTime" />
  <targetElements rdf:resource="#hrSWRunPerfCPU" />
  <inverseFormula
    rdf:resource="#hrSWRunPerfCPU->KernelModeTime,UserModeTime" />
  <expression>
    hrSWRunPerfCPU = (KernelModeTime +UserModeTime)*10;
  </expression>
</Formula>
<Element rdf:about="#KernelModeTime">
  <formula rdf:resource=
    "#KernelModeTime,UserModeTime->hrSWRunPerfCPU" />
  <mappedElements rdf:resource="#hrSWRunPerfCPU" />
</Element>
<Element rdf:about="#UserModeTime">
  <formula rdf:resource=
    "#KernelModeTime,UserModeTime->hrSWRunPerfCPU" />
  <mappedElements rdf:resource="#hrSWRunPerfCPU" />
</Element>
<Element rdf:about="#hrSWRunPerfCPU">
  <formula rdf:resource=
    "#hrSWRunPerfCPU->KernelModeTime,UserModeTime" />
  <mappedElements rdf:resource="#KernelModeTime" />
  <mappedElements rdf:resource="#UserModeTime" />
</Element>

```

5 Conclusions and further work

This paper has presented a method that uses ontology integration techniques to improve the network management information interoperability. Prior existing works deal with it in a very limited way, mainly based on recast translations. The method M&M takes into account the semantics contained in the information. With this, a common model can be generated to be used by a manager, independently of the underlying management domains. This fact will improve the management application development, which will be able to correlate data which until now did not have a direct association because they belonged to different models.

This method has been applied to a case study to validate it, obtaining interesting results. However, due to the needs of human supervision, it can take some time when integrating big specifications. In any case, this time will be less than doing this task manually.

A management system similar to that shown in Figure 7 can be developed based on these ideas. It will take advantage of this ontology-based approach, integrating all management models in a smart way and bearing in mind the semantics of the defined information. At the same time, generic gateways can be built that use the mapping ontologies generated with the method M&M to translate the information between management domains.

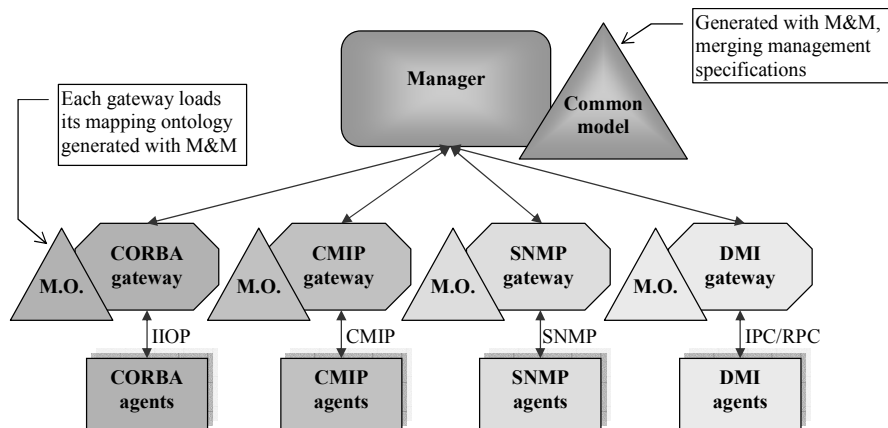


Figure 7. Architecture of a management system using this approach

Current works are also about the automation of M&M. An existing ontology tool [14] is being adapted to generate the mapping rules when merging two models. At the same time, additional heuristics to find candidates more easily are being studied to reduce the human interaction, including Natural Language Processing techniques to analyze the meaning of the descriptions.

6 References

1. Jorge E. López de Vergara, Víctor A. Villagrà, Julio Berrocal, "Semantic Management: advantages of using an ontology-based management information meta-model", *Proceedings of the HP Openview University Association Ninth Plenary Workshop (HP-OVUA'2002)*, Böblingen, Germany, June 2002.
2. Jorge E. López de Vergara, Víctor A. Villagrà, Julio Berrocal, Juan I. Asensio, Roney Pignaton, "Semantic Management: Application of Ontologies for the Integration of Management Information Models", *Proceedings of the Eighth IFIP/IEEE International Symposium on Integrated Network Management (IM'2003)*, Colorado Springs, Colorado, U.S.A., March 2003.

3. Jorge E. López de Vergara, Víctor A. Villagrà, Juan I. Asensio, Julio Berrocal, "Ontologies: Giving Semantics to Network Management Models", *IEEE Network*, special issue on Network Management, Volume 17, Number 3, May/June 2003.
4. R. Studer, V.R. Benjamins, D. Fensel, "Knowledge Engineering: Principles and Methods", *Data & Knowledge Engineering*, 25: 161-197, 1998.
5. Jun Shen, Yun Yang, "RDF-Based Knowledge Models for Network Management", *Proceedings of the Eighth IFIP/IEEE International Symposium on Integrated Network Management (IM'2003)*, Colorado Springs, Colorado, U.S.A., March 2003.
6. Emmanuel Lavinal, Thierry Desprats, Yves Raynaud, "A Conceptual Framework for Building CIM-Based Ontologies", *Proceedings of the Eighth IFIP/IEEE International Symposium on Integrated Network Management (IM'2003)*, Colorado Springs, Colorado, U.S.A., March 2003.
7. Natalya Fridman Noy, Mark A. Musen, "An Algorithm for Merging and Aligning Ontologies: Automation and Tool Support", *Proceedings of the Workshop on Ontology Management, Sixteenth National Conference on Artificial Intelligence (AAAI-99)*, Orlando, Florida, U.S.A., July 1999.
8. Deborah L. McGuinness, Richard Fikes, James Rice, Steve Wilder, "An Environment for Merging and Testing Large Ontologies", *Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR2000)*, Breckenridge, Colorado, U.S.A., April 2000.
9. Gerd Stumme, Alexander Maedche, "FCA-MERGE: Bottom-Up Merging of Ontologies", *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI 2001)*, Seattle, Washington, U.S.A., August 2001.
10. E-Commerce Integration Meta-Framework (ECIMF) Project, "ECIMF Semantic Translation tool", November 2001, available at <http://www.ecimf.org/software.html>
11. Alexander Maedche, Boris Motik, Nuno Silva, Raphael Volz, "MAFRA – A MAPPING FRAMework for Distributed Ontologies", *Proceedings of the Thirteenth European Conference on Knowledge Engineering and Knowledge Management (EKAW'02)*, Madrid, Spain, October 2002.
12. Dan Connolly, Frank van Harmelen, Ian Horrocks, Deborah L. McGuinness, Peter F. Patel-Schneider, Lynn Andrea Stein, "DAML+OIL (March 2001) Reference Description", W3C Notes, 18 December 2001.
13. J. Schönwälder and A. Müller, "Reverse Engineering Internet MIBs", *Proceedings of the Seventh IFIP/IEEE International Symposium on Integrated Network Management*, Seattle, Washington, U.S.A., May 2001.
14. N. F. Noy, M. A. Musen, "PROMPT: Algorithm and tool for automated ontology merging and alignment", *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI'00)*, Austin, Texas, U.S.A., July 2000.