Ontology-based Semantics for Composable Autonomic Elements

John Keeney, Kevin Carey, David Lewis, Declan O'Sullivan, Vincent Wade
Trinity College Dublin
Knowledge and Data Engineering Group
Computer Science Department, College Green, Dublin 2, Ireland
{John.Keeney|Kevin.Carey|Dave.Lews|Declan.OSullivan|Vincent.Wade}@cs.tcd.ie

Abstract

The complexity of modern communication networks requires an autonomic approach, where elements exhibit a degree of self-management which when combined provide a level of self-management for the network as a whole. The heterogeneity of elements however prompts a knowledge driven approach to their definition, composition and management in order to address problems of semantic interoperability. This paper proposes a semantic service based approach to the definition of elements in an autonomic network in order to enable ontological reasoning in support of composable self-management functions.

1 Introduction

The management of computing and communications systems has traditionally been a skilled human task, so 'self-management' is only appropriate if it is overseen or governed in a manner understandable to a human controller. Autonomic communications systems are adaptive networks, the adaptive behaviour of which is governed by human-specified goals and constraints on how the services provided by the network should behave.

The self-management of network elements requires dynamic mapping of human management goals to enforceable policies across a system, with the adaptive network elements reacting to changing context. However, this adaptivity must operate within constraints set by human-specified policies. Accurately mapping these high level policies or governance directives, down to low level adaptation and control policies for individual heterogeneous functional elements poses a challenge network administration and one for which automated solutions remain elusive. It is further complicated by the mapping typically having to occur in the context of a specific service chain or flow within more richly connected network of managed components

This paper introduces a Service-Oriented approach that presents a model of constrainable adaptivity for heterogeneous network management functions. In this model, resources are managed as composable services, called Adaptive Service Elements (ASE), containing inbuilt application-specific adaptivity in its use of sub-services and its subscription to relevant context information streams. In this manner, services can be composed into value chains and workflows while also exposing an elemental resource management view which can form part of an end-to-end resource management activity.

This paper also introduces how ontology-based semantics help address conceptual heterogeneity between services and context and provide a reasoning framework for policy refinement.

2 Semantic Services

Ontology-based semantics [berners-lee], proposed by the Semantic Web initiative, help solve some of the problems of heterogeneity and runtime discovery of service capabilities. Web Service Definition Language (WSDL), a standardised service description language, describes the functional aspects of services and so enables the definition of service operations along with their input and output parameters. However, a richer semantic language is needed in order to reason about services that must be discovered, composed or invoked dynamically. The OWL-based Web Service Ontology (OWL-S) [owls02] uses ontology-based semantics to enhance such web service descriptions. It uses description logic based ontologies, specified in the Web Ontology Language (OWL) [owl], and emerging semantic rule languages to define the Inputs, Outputs, Preconditions and Effects of a service (often abbreviated to IOPE), and in addition describes the resources used by that service. OWL-S provides an unambiguous, computer-interpretable semantic description of a service by providing rich definitions of the IOPEs of a service's operations, as well as a rich set of control specifications for linking constituent services. Through this semantic approach, inference engines (e.g., AI planners and matchmakers) are enabled to automate the discovery, composition, invocation, and monitoring of services [mclraith] despite the use of separately authored ontology models for describing IOPEs and resources.

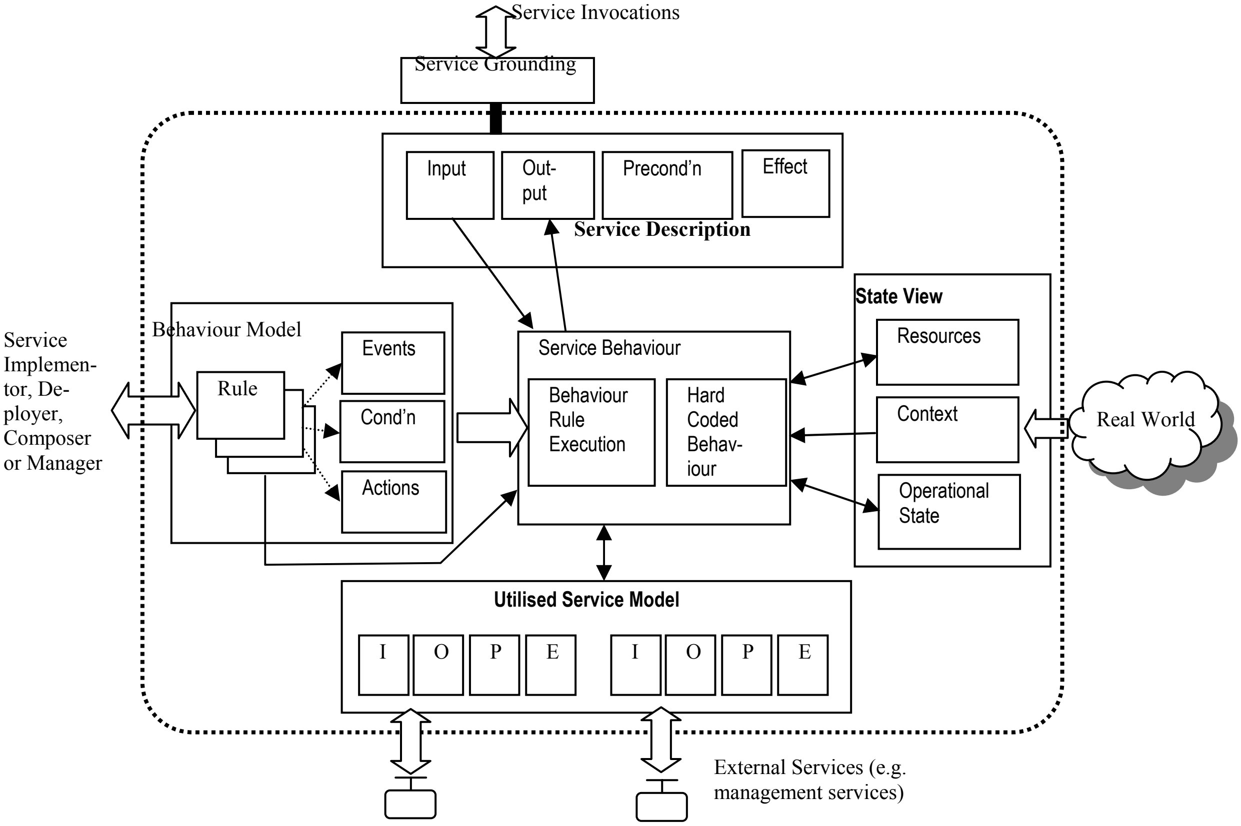


Figure 1: Adaptive Service Element Reference Architecture

3. Semantics of Autonomic Elements

Using a service-oriented approach to the management of autonomic network elements, a service interface is used to provide access to a specific set of resources, where the resources are controlled by the implementation of the service, either solely or shared with other service implementations. We model a service and its behaviour using the abstract concept of an Adaptive Service Element [lewis04] (See Figure 1). This offers a specific service, the behaviour of which is aware both of its local operational context and the characteristics of the network of which it is a part. It is aware of and controls a specific set of resources, which may be modelled as a further set of services.

The adaptive behaviour of an ASE will need to be managed to reflect the goals and requirements of the service users, those people or agents responsible for the service's resources, and the managers who oversee the operation of the network being managed. This management is performed by providing behavioural rules to the adaptive service element. These rules dictate the element's behaviour within the constraints provided by the element's developers, either human designers or automated agents that generate service compositions. Adaptation policy rules, specified as part of the service behaviour model, can be set by the service administrator to adapt how the particular service makes use of particular resources. In this way, the management of such a service-oriented system is achieved by policies local to service, rather than by policies that relate generally to the underlying resources. Such policies can be specified as action, goal or, utility rules. Overall, the responsibility falls on the service developer to expose, via the semantic service specification and behaviour model, all the adaptable interactions between the service and the resources it uses and manages.

Based on this, an ASE is characterised by: a service description; a model of the state observable by the ASE; a description of the services of which it makes use; and a rule-based model for describing and restricting its behaviour component's managed behaviour can be seen as a rule-based automaton. Each service element will also require a OWL-S grounding for each target platform technology that will use it.

Policy refinement is the decomposition of policies relevant to a composite system into a set of policies that are executed in its constituent parts, thereby implementing the behaviour intended by the overall system level policy. In order for even semi-automated policy refinement to be successful, it is necessary to have access to semantic information about both the high level policy and the service being managed. To enable adaptive systems to process such semantics automatically we adopt ontology based semantics as a means of describing constraints on an adaptive service element's behavioural rules in a machine intelligible form. The expression of these constraint semantics is eased by having the semantics of services and the operational context also expressed in an ontological format.

4 Semantics for Autonomic OSS

Within the Semantic Web initiative it has been widely observed that ontological reasoning techniques will only become beneficial once a sufficiently large number of available services have been semantically marked-up. Similarly in the context of autonomic management, ontology-driven policy refinement will only be of use for autonomic systems once services and networks possess ontological representations.

To arrive at a situation where ontology-based semantics can be fruitfully employed in network operations, we must first move from the current state of the art in communications management technology used in Operational Support Systems (OSS). The predominant paradigm in network management has been the manager-agent model. This models management interface functionality in a fine-grained object-oriented manner, where management functionality is provided by get and set operations on object attributes, and depending on the model used is supplemented by object-level actions and notifications. Functional interface models are defined in terms of Management Information Base (MIB) specifications. Here, the OSI Management and Internet Management represent the two main standards bodies, using the GDMO and SMI languages respectively. Both of these languages, though being potentially generic profiles of ASN.1, were shaped in their usage by the features of the protocols that accompanied them, CMIP and SNMP respectively.

In the 1990s the Distributed Management Task Force defined the Common Information Model schema which was a principled attempt to define management information models for the manager-agent paradigm, but in a way that was independent from the protocol used. This proved successful, quickly becoming a focus for management information modelling standardisation effort, especially in the enterprise management sphere, with support added for a number of protocol bindings including DCE, XML/HTTP and LDAP. The modelling approach was highly object-oriented, yet also incorporated a number of ontological modelling concepts, such as making associations first class concepts with domain and range bindings to classes and allowing class and instance definitions to be freely mixed.

More recently Jorge de Vergara and Victor Villagra [deVergara] have show directly the value of modelling management information models in OWL, and how this can be used to ease the interoperation between models originally conceived in different MIB languages, i.e., GMDO, SMI, CIM.

In parallel, the engineering of service and business layer OSSs for the telecommunication market began to adopt the service-oriented and n-tier component architectures that had come to dominate enterprise computing. At the forefront of attempts to reach industry agreement on modelling such architectures for communications management was the TeleManagement Forum's NGOSS initiative [fleck]. This is attempting to stimulate an open market in telecoms business software component by forming agreements on management information exchanged between business processes and service definitions, via which inter-process invocations can be made. The former encompasses network and element level MIB information as well as service and business level information typically captured in corporate databases. Such business objects also increasingly become the subject of business-to-business e-commerce agreements, e.g. ebXML. This has a natural synergy with the enterprise management model of the DMTF, and the two organisations are now collaborating closely on information modelling. The models for inter-process invocation, termed contracts, are defined in a native XML binding [tmf053] that includes the usual input and outputs as well as preconditions and effects and other service component lifecycle information, e.g. vendor data, deployment setting etc.

It can be seen therefore that the emerging understanding of how semantic web ontology languages may assist in the semantic interoperability of management information models should be naturally reflected in the application of OWL-S to the definition of business application services for the OSS domain. In particular, the technology neutral approach taken in the NGOSS initiative would seem ripe for an ontological approach, provided suitable methodologies and tools emerge [duke]. For this reason our current investigations are moving in this direction, whereby we attempt to re-model existing OSS service components with OWL-S in an attempt to better understand the specific benefits of semantic interoperability and ontology-based conceptual reuse in the OSS software engineering domain.

5 Semantic Reasoning for Autonomics

Though the Adaptive Service Element reference model represents our target architecture for future autonomic communications networks, we acknowledge the need to take a number of exploratory steps in reaching it. This section outlines a number of specific directions currently being explored, and gives initial results where available.

5.1 Dynamic Service Composition

Artificial Intelligence (AI) planning is one technique that is receiving increased attention as a solution for automated service composition and automated adaptivity control. AI planning techniques can automatically generate composite service plans consisting of simple sequence of actions. Each action can be supported by service invocations, given a set of required goals, a set of possible actions and a description of the initial state of the system. AI planning seeks to represent a relevant part of the world in terms of various states and possible changes that can be made to those states. For example, one branch of planning known as Situational Calculus classifies the functional properties of a service as Inputs, Outputs (states of knowledge of the user) and Preconditions and Effects (world states), which may be available in a semantic description of the service. This rigid approach to world representation allows the usual suite of AI techniques to be applied to a huge range of problems, including automatic service composition and automatic service adaptation. In a further approach, used only for adaptive service composition in [higel03], an analysis of the durative characteristics of services is used to compose services in an intelligent manner. The ASE model will focus on the use of more sophisticated approaches to driving the AI planning mechanisms used, not just to compose services, but to manage the adaptive behaviours of network elements, handled in a service-oriented manner.

5.2 Semantic Management Services

Clearly before we can make good use of AI planning techniques for autonomic communications we need a sufficiently rich set of services from which to compose new services. The ASE promotes a service oriented approach to developing new application components with a policy management interface that make them suitable for use in an autonomic framework. This requires a development approach, where service and management features are closely coupled at design time. As this approach is not currently widespread, we envisage a long period before such a sufficient large population of such components will have been developed to make AI planning viable as an effective adaptive technique.

```
<CLASS SUPERCLASS="CIM_EnabledLogicalElement" NAME="CIM_LogicalDevice">
…
<METHOD CLASSORIGIN="CIM_LogicalDevice" NAME="Reset" TYPE="uint32">
<QUALIFIER TRANSLATABLE="true" NAME="Description" TYPE="string"> <VALUE> Requests a reset … </VALUE> </QUALIFIER>
</METHOD>
…
</CLASS>

<CLASS SUPERCLASS="CIM_LogicalDevice" NAME="CIM_Printer">
…
<PROPERTY CLASSORIGIN="CIM_Printer" NAME="MaxCopies" TYPE="uint32">
   <QUALIFIER TRANSLATABLE="true" NAME="Description" TYPE="string"> <VALUE>The maximum ……</VALUE> </QUALIFIER>
</PROPERTY>
<PROPERTY CLASSORIGIN="CIM_Printer" NAME="PrinterStatus" TYPE="uint16">
   <QUALIFIER TRANSLATABLE="true" NAME="Description" TYPE="string"> <VALUE>Status information for a Printer …. </VALUE></QUALIFIER>
   <QUALIFIER NAME="ValueMap" TYPE="string"> … <VALUE>1</VALUE><VALUE>2</VALUE> … </QUALIFIER>
   <QUALIFIER TRANSLATABLE="true" NAME="Values" TYPE="string">…<VALUE>Idle</VALUE><VALUE>Printing</VALUE></QUALIFIER>
   <QUALIFIER NAME="MappingStrings" TYPE="string"> … <VALUE>MIB.IETF|Printer-MIB.hrPrinterStatus</VALUE> </QUALIFIER>
</PROPERTY>
…
</CLASS>
```

Figure 2: CIM printer data and methods in XML format

```
…
<owl:FunctionalProperty rdf:ID="CIM_Printer_MaxCopies">
   <rdfs:domain rdf:resource="#CIM_Printer"/>   <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
   <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string">"The maximum …"</rdfs:comment>
   <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>   <rdfs:label>CIM_Printer:MaxCopies</rdfs:label>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="CIM_Printer_PrinterStatus">
   <rdfs:domain rdf:resource="#CIM_Printer"/>   <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
   <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string">"Status information for a Printer …"</rdfs:comment>
   <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>   <rdfs:label>CIM_Printer:PrinterStatus</rdfs:label>
</owl:FunctionalProperty>
…
```

Figure 3: CIM printer data in OWL format

However, to initiate exploration of this possibility and to study in detail the means by which existing network semantics can be captured and used, we examine the use an ASE management interface that resembles more a semantic version of current manager-agent oriented interfaces, rather than the target policy-oriented interfaces. The approach taken is to use the algorithms described in [deVergara] to extract the class and property information contained in existing information models and then to integrate them with management service models based on OWL-S.

Here we take, as an example, a segment from the DMTF CIM information model for a printer, figure 2, and map to OWL, figure 3. From the information available by mapping the CIM management interface to an ontological format, any access to this data can be reasoned about, with the possibility of the management interface being automatically created. For example, from the excerpts from the Printer Device MOF above, and taking into account the default values of qualifiers not shown, a number of conclusions can be inferred from the ontology, e.g.:

- An operation to read the properties PrinterStatus and MaxCopies are required, but operations to set them are not required since “readonly” is the default for properties.
- Operations to return a descriptive string for each property is required.
- The method reset() is required for the CIM_Printer management interface, since methods propagate to subclasses by default.

This knowledge refers to how one interacts with the model, rather than the semantics of its informational aspects. This can thus be better expressed in OWL-S format, allowing the dynamic creation of such a semantic management interface. A full OWL-S description of this information model segment is outlined in figure 4. It would include a core definition of the process (figure 5) to allow an ontological reasoner, e.g. an AI planner, to reason about the inputs and outputs of such operations, the preconditions and effects of the operations, the types of operations allowed, or how the operations can be composed. A service profile would allow this service to be advertised, e.g. using UDDI, for use in a semantically driven service discovery process. The OWL-S grounding model could then be used in automated invocation of the management service. The example in figure 7 indicated a WSDL grounding, but a grounding to the specific XML and HTTP bindings defined by the DMTF Web Based Enterprise Management standards could be developed and used here equally.

```
<service:Service rdf:ID="CIM_Printer_Service">
   <service:describedBy rdf:resource="http://…/ …  #_Process"/>
   <service:presents rdf:resource="http://…/ … #_Profile"/>
   <service:supports rdf:resource="http://…/ … #_Grounding"/>
</service:Service>
…
```

Figure 4: Top level OWL-S definition for the CIM_Printer Service

```
<process:AtomicProcess rdf:ID="CIM_Printer_Class_getPrinterStatus">
 <process:hasResult>
  <process:Output rdf:ID=" CIM_Printer_Class_getPrinterStatusReturn_OUT ">
     <process:parameterType>”#CIM_Printer_PrinterStatus” </process:parameterType>
  </process:Output>
 </process:hasResult>
</process:AtomicProcess>
…
<process:AtomicProcess rdf:ID="CIM_Printer_Class_getMaxCopies">
  <process:hasResult>
   <process:Output rdf:ID="CIM_Printer_Class_getMaxCopiesReturn_OUT ">
      <process:parameterType>”# CIM_Printer_MaxCopies” </process:parameterType>
   </process:Output>
  </process:hasResult>
</process:AtomicProcess>
…
```

Figure 5: The OWLS Processes that make up the CIM_Printer Service

```
<profile:Profile rdf:ID="_Profile">
  <profile:hasOutput rdf:resource="http://…/_ProcessModel# CIM_Printer_Class_getPrinterStatusReturn_OUT "/>
  <profile:hasOutput rdf:resource="http://…/_ProcessModel# CIM_Printer_Class_getMaxCopiesReturn_OUT "/>
  <profile:hasOutput rdf:resource="http://…/_ProcessModel# CIM_Printer_Class_reset_OUT "/>
…
</profile:Profile>
```

Figure 6: The OWLS outputs profile of the CIM_Printer Service

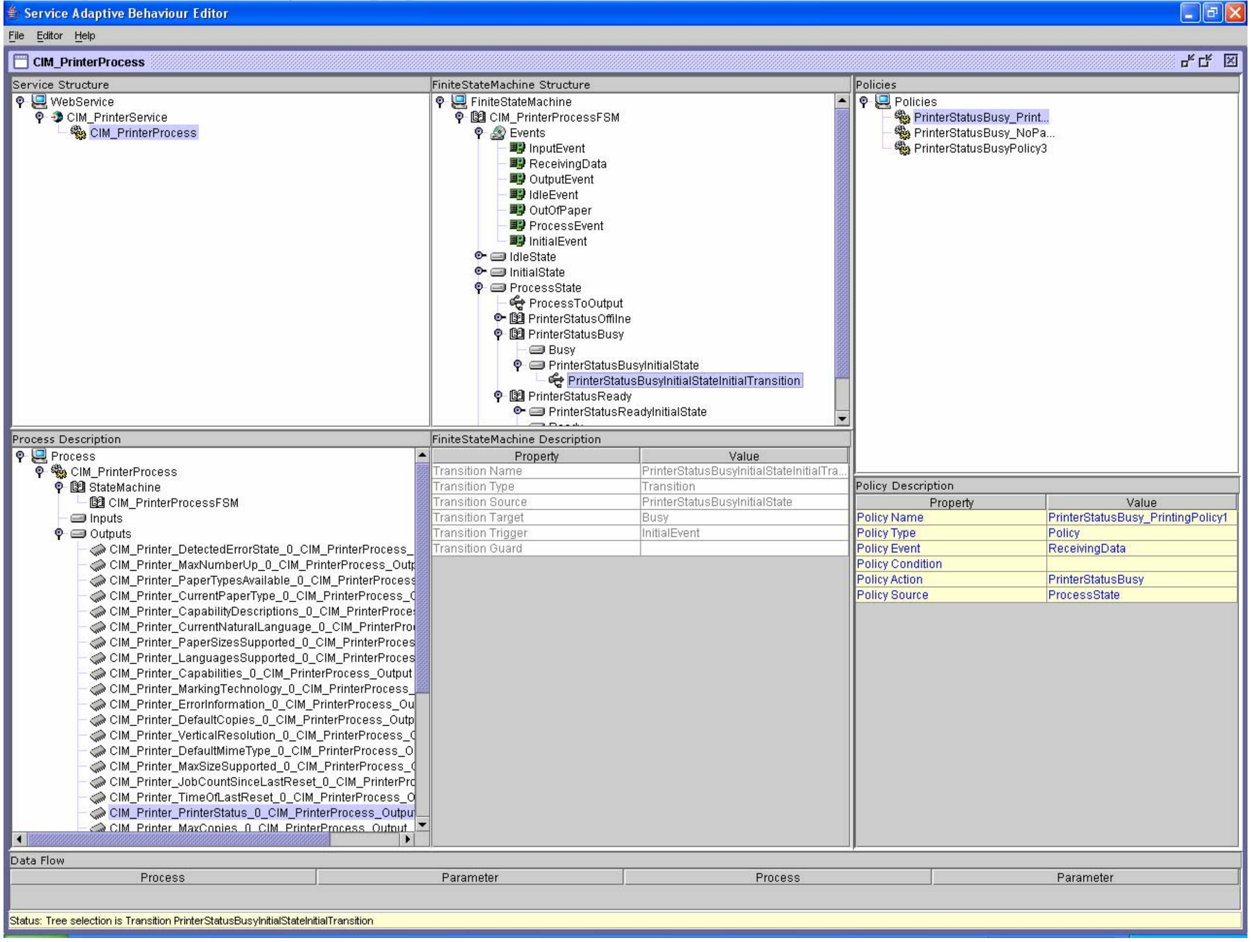
```
<grounding:WsdlGrounding rdf:ID="_Grounding">
  <service:supportedBy rdf:resource="http://asdfasdf/_Service#_Service"/>
  <grounding:hasAtomicProcessGrounding rdf:resource="#WSDLGrounding__getPrinterStatus"/>
  <grounding:hasAtomicProcessGrounding rdf:resource="#WSDLGrounding__getMaxCopiesReturn"/>
  <grounding:hasAtomicProcessGrounding rdf:resource="#WSDLGrounding__reset"/>
…
</grounding:WsdlGrounding>
```

Figure 7: How the CIM_Printer Service would be grounded by some service described by WSDL

5.3 Policy-based Management for Composite Services

When adaptive services are composed, inevitably the behaviour rule sets grow and become unmanageable. In the policy refinement approach discussed in [carey04] adaptive behaviour rules (high level policies) can be automatically described for a composite service element, specified as finite state machine transitions, which are automatically refined into state transitions for the sub-finite state machines describing the ASE's adaptive behaviour. Here, the use of component behaviour ontologies based on finite state machines can be used to expose just a selected subset of behaviour for policy-based management purposes. We have conducted some preliminary prototyping of a tool for modelling finite state machines using behavioural concept expressed in OWL, e.g. the CIM Printer MIB used in the previous section as depicted in figure 8.

However, as can be seen from the management of complex adaptive systems such as network management systems [murray05], such a discrete state-based model is not sufficient. For an autonomic system to manage a network of adaptive network elements, a more expressive approach is needed to ensure that adaptivity is constrained in a manner where the network operates within an envelope of acceptable behaviour within a certain behaviour space [dobson04], rather than the fixed and restrictive manner described.



Property	Value
Transition Name	PrinterStatusBusyInitialStateInitialTra...
Transition Type	Transition
Transition Source	PrinterStatusBusyInitialState
Transition Target	Busy
Transition Trigger	InitialEvent
Transition Guard	

Property	Value
Policy Name	PrinterStatusBusy_PrintingPolicy1
Policy Type	Policy
Policy Event	ReceivingData
Policy Condition	
Policy Action	PrinterStatusBusy
Policy Source	ProcessState

Status: Tree selection is Transition PrinterStatusBusyInitialStateInitialTransition

Figure 8: Automatic creation of management policies based on OWLS specifications of services

5.4 Semantic Interoperability

These proposed service-oriented ASEs allow interaction through well defined service interfaces, thereby allowing an adaptive communications framework to be constructed from elements sourced from any number of developers. There will, however, be a need for semantic interoperability through the resolution of semantic mismatches which will be inevitable due to the nature of differing vendor developments. It is expected however that resolution of these conflicts will be supported through the use of tools and processes which support the creation/discovery of mappings between ontologies. Substantial research has been ongoing into the area of semi-automatic techniques for mappings [osullivan03] but very little research has been undertaken into its applicability to autonomic systems. In particular we will study the extent to which semi-automated mapping approaches to semantic interoperability are sufficient in fulfilling the needs of autonomic systems and we will develop appropriate solutions.

5.5 Knowledge Delivery Network

In an autonomic computing environment, ASE's may be adaptable, but they their adaptation must be driven by both local context and network context. However, difficulties arise when heterogeneous elements must provide possibly complex end-to-end service provision chains over an adapting network topology. This heterogeneity leads to increased human costs to manage connections for information exchange. This can be alleviated by the provision of an active Knowledge Delivery Network to replace the standard passive information retrieval model. A Semantic Query Based Network is described in [lewis04] that uses a publish/subscribe paradigm from Content Based Networking to support the dissemination of ontologically defined knowledge. Such a model can be further expanded to act as a suitable Knowledge Delivery Network, with the semantic interoperability effort invested in the delivery network for use not just for managing the network but for other applications using the autonomic communications framework. Such an approach raises several issues, including ensuring suitable access control in multi-organisational setting, and where semantic interoperability functions may best be located.

6. Conclusions and Further Work

To conclude, this paper proposes a semantic service based approach to the definition of elements in an autonomic network in order to enable ontological reasoning in support of self-management functions. We are currently working on examining the use of semantics for various parts of the ASE reference mode. Specifically we are using OWL classes and properties derived from existing management information models as the core concepts for defining ASE state, both its resources and context, which is then also used in defining finite state machine definitions for the service adaptive behaviour. This adaptive behaviour model can then be used as the basis for defining run-time policies constraining the behaviour of the ASE. We are also using this MIB derived OWL model as the basis for input and output of OWL-S services providing management capabilities. This offers management capability in the more traditional manager-agent mode of interaction, but also offers the possibility of auto-generating semantic management services from MIB definition in a form that can be fed into an AI planner, so that composite management operations can be generated dynamically.

In addition we are using OWL-based management semantics as the basis for examining run-time semantic interoperability both for semantic service invocations and for content-routed semantic notifications in the knowledge delivery service.

Acknowledgments

This work has been partly sponsored by Science Foundation Ireland through the Centre for Telecommunications Value Chain Research, and partly by the Irish Higher Education Authority through the M-Zones programme.

References

[abelson *et al.*, 1985] Harold Abelson, Gerald Jay Sussman, and Julie Sussman. *Structure and Interpretation of Computer Programs.* MIT Press, Cambridge, Massachusetts, 1985.

[berners-lee] Berners-Lee, T., Hendler, K., Lassila, O. (2001), *The Semantic Web*, Scientific American, pp 35-43, Issue 284 (3), 17th May 2001

[carey04] Carey, K., Lewis, D., Higel, S., Wade, V., *Adaptive Composite Service Plans for Ubiquitous Computing*, 2nd International Workshop on Managing Ubiquitous Communications and Services (MUCS2004), Dublin, Ireland, November 2004.

[deVergara] de Vergara, J.E.L., Villagrá, V.A., Berrocal, J., *Applying the Web Ontology Language to management information definitions*, IEEE Communications Magazine, Vol. 42, Issue 7, July 2004, pp. 68-74. ISSN 0163-6804

[dobson04] Dobson, S. and Nixon, P., *More principled design of pervasive computing systems*. In Proc Engineering for Human-Computer Interaction and Design, Specification and Verification of Interactive Systems.Hamburg, Germany, July 2004.

[duke] Duke, A., Davies, J., Richardson, M., Kings, N., *A Semantic Service Oriented Architecture for the Telecommunications Industry*, in proc of Intelligence in Communications Systems, (INTELLICOM 2004), Bankok, Thailand, Nov 2004, LNCS 3283, pp 236-245

[fleck] Fleck, J, *An Overview of the NGOSS Arhcitecture*, TeleManagmeent Forum Whitepaper, May 2003

[higel05] S. Higel, T. O'Donnell, D. Lewis, V. Wade, *Towards an Intuitive Interface for Tailored Service Compositions*, Proc.

Distributed Applications and Interoperable Systems, 4th IFIP WG6.1 International Conference, DAIS 2003, Paris, France, November 17-21, 2003, LNCS 2893. pp 266-273

[lewis04] D. Lewis, T. O'Donnell, K. Feeney, A. Brady, V. Wade, *Managing User-centric Adaptive Services for Pervasive Computing,* Proc. IEEE International Conference on Autonomic Computing (ICAC'04), May 17th-18th 2004, New York, USA.

[mclraith] McIlraith, S.A., Son, T.C., Honglei Zeng, H. (2001), *Semantic Web Services*, IEEE Intelligent Systems, 16(2), March/April 2001

[murray05] K. Murray, D. Pesch, *Revenue Optimisation and User Prioritisation using Call Admission Control Strategies in Multi-service 4G Cellular Wireless Networks*, in Proc. European Wireless 2005, Cyprus, April 2005

[osullivan03] D. O'Sullivan, D. Lewis, *Semantically Driven Service Interoperability for Pervasive Computing*, Proc. 3rd ACM International Workshop on Data Engineering for Wireless and Mobile Access, San Diego, USA, 19 Sept 2003, ACM Press.

[owl] World Wide Web Consortium (W3C), *Web Ontology Language (OWL)*, www.w3.org/2004/OWL/, Visited Mar 2005

[owls02] The DAML Service Coalition, *OWL-S: Semantic Markup for Web Services.* http://www.daml.org/services/, October 2002

[tmf053] TeleManagement Forum, *NGOSS Technology Neutral Architecture Release 4.0*, January 2004,