Composing User Network Operation Services Using Web Service Composition Techniques

Nobuhide Nishiyama, Kenichi Nishikawa, Fumihiko Ito
NTT Access Network Service Systems Laboratories
NTT Corporation
Tsukuba, Ibaraki, Japan
{nisiyama, nisikawa, f.ito}@ansl.ntt.co.jp

Yasuhiro Suzuki
Network Business Headquarters
NTT East Corporation
Chiyoda-ku, Tokyo, Japan
suzuki.yasuhiro@rdc.east.ntt.co.jp

***Abstract*—This paper proposes a method for composing user network operation services, using Web service composition techniques. We define two types of common operation services (COSs) for user network resources. Atomic COSs do not rely on other COSs, and each of them is provided by an elemental function of each piece of network equipment. Composite COSs are composed of other atomic or composite COSs, and provide users with practical operation services. Network resources and COSs are described in semantic languages such as OWL and OWL-S, which realize Web service composition, and composite COSs are composed based on those descriptions and composition rules. Examples of composite COSs are services for QoS control between each gateway and end points, which are composed of atomic COSs for control of each piece of network equipment. This approach provides users with automatic composition of practical network operation services in accordance with each user network configuration.**

Keywords

user network, home network, network management and control, service composition

I. INTRODUCTION

Broadband access network infrastructures, such as asymmetric digital subscriber lines (ADSL) and fiber to the home (FTTH), have spread rapidly to user areas. Moreover, home networks have been attracting attention due to the rapid spread of PCs and the appearance of information appliances. Premises networks are also now being constructed in locations such as condominiums, offices and schools. This progress has generated a variety of network services, including the delivery of high quality movies and music, and real-time audio-visual communication services.

If we are to achieve end-to-end service quality management in these environments, we must manage and operate user network resources. We expect user networks to be operated under the following conditions.

- The pieces of user network equipment have various network protocols (such as Ethernet, Wireless LAN, Bluetooth [1] and IEEE1394) and various functions (such as those for QoS and security control). Besides, they are often provided by various vendors.
- Users usually reconstruct user networks by altering the network configuration and by adding equipment with new protocols or from new vendors.

To achieve user network operation without the need to take protocols, vendors and network reconstruction into account, we have studied a method for providing common operation services (COSs) for operation of each piece of equipment, by integrating a variety of proprietary operation services [2].

However, considering that many of the end users are inexperienced in network operation, it is difficult for them to operate their networks themselves by extracting and executing required COSs provided by each piece of equipment. To make it possible for them to operate their networks, it is necessary to study a method for automatic composition and execution of COSs in accordance with users' demands entered via easy-to-use user interfaces for network operation.

In this paper, aimed at providing end users with such user network operation environments, we propose a method for automatic composition of COSs, based both on the network operation studies and on Web services studies.

The automatic composition of Web resources providing particular services based on semantic information has been studied in relation to Web services [3]. This Web service composition seems to be analogous with that for COSs for user network operation described above. Based on this consideration, we apply a method for service modeling and service composition studied for Web services to user network operation services.

II. FUNCTIONS OF USER NETWORK MANAGEMENT AND OPERATION SYSTEM

To make it possible to realize composition of a variety of COSs for user network operation, we propose that a user network management and operation system (UNM) is installed in a piece of equipment such as a home server and a gateway in each user network. Figure 1 shows the architecture of a UNM. The functions of the UNM are described below.

- The COS library

  The COS library provides COSs for operation of user network resources, such as setting/obtaining the configuration of each piece of equipment. There are two types of COSs. Atomic COSs are services that do not rely on

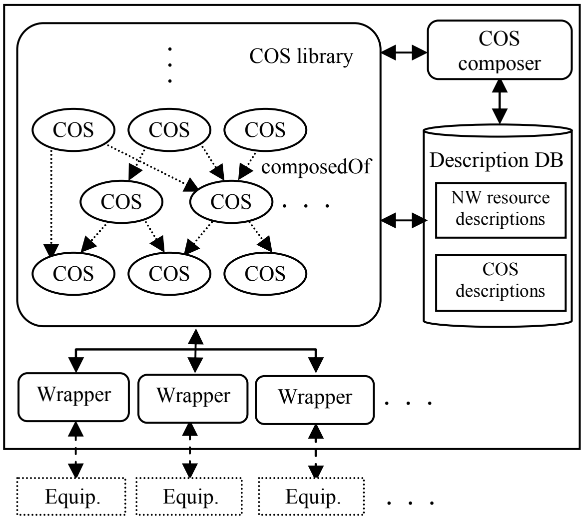

Figure 1. UNM architecture

other COSs, and each of them is provided by an elemental function of each piece of network equipment. An example of an atomic COS is the SetMinBand service that realizes configuring the minimum assured bandwidth of each egress queue. When the SetMinBand service is called, it sends the request to each piece of equipment using the appropriate wrapper module. On the other hand, composite COSs are services composed of other atomic or composite COSs, and provide users with practical operation services. Examples of composite COSs are services for QoS control between a gateway and end points [4].

- The description DB

As regards composing COSs automatically, semantic information about network resources and COSs plays an important role. The description DB manages such semantic descriptions of network resources and COSs.

Network resource descriptions include class and instance descriptions. Network resource class descriptions include class definitions of network resources, and are sharable among user networks. On the other hand, network resource instance descriptions are the descriptions of the instantiated network resources, and include information on the network configuration and the COSs provided by them.

COS descriptions also include class and instance descriptions. COS class descriptions include class definitions of atomic or composite COSs and, in the case of composite COSs, composition rules. They are sharable among user networks, as network resource class descriptions are. On the other hand, COS instance descriptions are the descriptions of instantiated COSs in each user network due to the class descriptions, and include information on the operation processes.

There are several languages for description of semantic information. For network operation, languages such as GDMO [5] and SMIv2 [6] are proposed. On the Web, Semantic Web languages, which include languages such as OWL [7] and OWL-S [8], are proposed. Comparing these languages, Semantic Web languages can realize the richest description including axioms and inference rules necessary for service composition. They can also realize description of service composition. Based on this consideration, we describe network resources and COSs using Semantic Web languages, by newly adding descriptions such as class definitions of network resources and COSs, relationships between a network resource and a COS, and relationships among network resources.

- The COS composer

The COS composer generates composite COSs. It derives and generates COS instance descriptions of composite COSs, based on the network resource descriptions and COS descriptions. It then generates COSs based on the generated descriptions.

- The wrapper modules

Each wrapper module converts requests for atomic COSs realizing setting/obtaining the configuration of each piece of equipment to proprietary operation commands (which use protocols such as SNMP and Telnet). Each module is downloadable as a result of the addition of a new type of equipment, due to the existing network management information such as MIBs. Each module also registers the network resource and COS instance descriptions. The acquisition and execution of the appropriate module can be realized by using, for example, an OSGi Service Platform [9].

Due to the architecture described above, the COS composer can compose various composite COSs based on the semantic information described in the network resource and COS descriptions, without the need to consider the proprietary operation services provided by each piece of equipment.

After this, we study a method for generating COSs for QoS control between each gateway and the end points, as examples of composite COSs. Here, we assume that the COS composer only generates composite COS instance descriptions, and they are executed by the interpreter.

III. USER NETWORK RESOURCE AND COS MODELING

A. *User network resource modeling*

Several network modeling methods are proposed in ITU-T [10, 11, 12] and TMF [13, 14]. We proposed an abstract model for access networks [15] and user networks [4], with reference to these propositions. Here, we redefine the user network model, as shown in Figure 2.

User networks are composed of several network elements, including terminal/network/gateway equipment. They also include links, connections, and termination points for each link and connection. Each user network is terminated with a piece of terminal or gateway equipment.

Equipment represents an equipment entity. That is, it represents terminal equipment (such as a PC, or an information

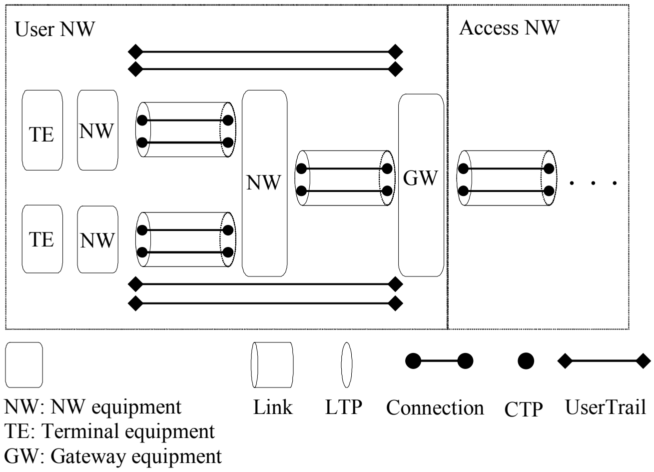

Figure 2. User network resource model

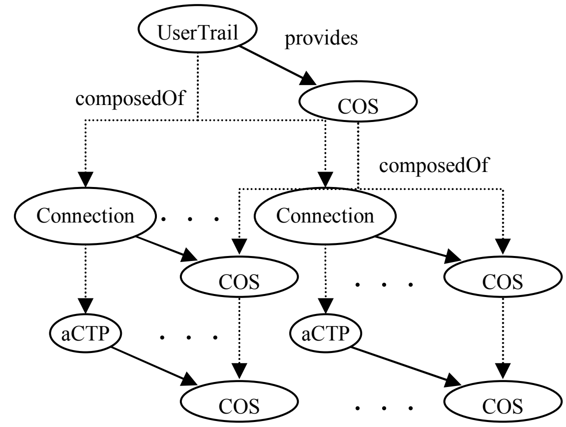

Figure 3. User network resource and COS model

appliance), network equipment (such as an L2-SW, an L3-SW and bridging equipment between Ethernet and Bluetooth, or Ethernet and wireless LAN) or gateway equipment. Each piece of Equipment includes several Link Termination Points (LTPs).

LTP represents a physical termination point entity and terminates a Link. Each LTP includes several Connection Termination Points (CTPs).

CTP represents a logical termination point entity and terminates a Connection. We assume that each CTP corresponds to each egress queue.

Link represents a physical link entity. Each Link has an LTP at each end, and includes several Connections.

Connection represents a logical link connection entity. Each Connection transports packets with the same quality control method. Each Connection has a CTP at either end.

UserTrail represents a serial connection of Connection entities, such as the connection between two edges (gateway or network equipment accommodating terminals).

To discuss COSs for QoS control between each gateway and the end points, we assume that UserTrails are the resources between each gateway and the end points, and they provide those composite COSs for QoS control.

B. COS modeling based on the network resource model

We study the COS modeling based on the network resource model proposed above. Figure 3 shows part of the resource and COS model, after execution of service composition for UserTrail operation.

Each COS is provided by each of the user network resource entities described above. Each COS provided by a UserTrail is composite and composed of COSs provided by component Connections. Each COS provided by a Connection is also composite and composed of an atomic COS provided by a component aCTP ("aCTP" means the upper CTP). Instance descriptions of COSs provided by CTPs are provided by each wrapper module. Instance descriptions of composite COSs provided by UserTrails are generated based on those of atomic COSs provided by component aCTPs.

Some of COSs provided by the resources shown in Figure 3 are described below.

- COSs for control of QoS parameters

  Each resource can provide COSs such as SetMinBand, SetMaxBand and SetPriority. The purpose of these COSs is to configure the minimum assured bandwidth, maximum bandwidth and priority realized by each resource while transporting packets, respectively.

- COSs for control of traffic permission

  Each resource can provide COSs such as AddFlow and AddClass. The purpose of these COSs is to configure per-flow and per-class traffic permission, respectively.

- COSs for control of the marking configuration

  Each resource can provide COSs such as SetMarking. The purpose of these COSs is to configure packet marking such as DSCPs and 802.1p priority tags.

Figure 4 and Figure 5 show part of an example of a network resource instance description and an instance description of a composite COS for operation of a UserTrail, respectively. They are provided in OWL and OWL-S, which are ones of semantic languages on the Web. The OWL language provides descriptions of classes, instances and relationships between them. The OWL-S language provides descriptions for Web service discovery, execution, composition and interoperation.

The description shown in Figure 4 represents the fact that the resource named "TR1" is an instance of the UserTrail class, and composed of the resources named "CON1" and "CON2", which are instances of the Connection class. It also represents the fact that TR1 provides the COS named "TR1_AddFlow",

```
<owl:Thing rdf:ID="TR1">
  <rdf:type rdf:resource="#UserTrail"/>
  <ex:composedOf>
    <ex:components rdf:parseType="Collection">
      <ex:nwResource rdf:about="#CON1"/>
      <ex:nwResource rdf:about="#CON2"/>
    </ex:components>
  </ex:composedOf>
  <ex:provides rdf:resource="#TR1_AddFlow"/>
</owl:Thing>

<owl:Thing rdf:ID="CON1">
  <rdf:type rdf:resource="#Connection"/>
  <ex:composedOf rdf:resource="#CTP1"/>
  <ex:provides rdf:resource="#CON1_AddFlow"/>
</owl:Thing>

<owl:Thing rdf:ID="CON2">
  <rdf:type rdf:resource="#Connection"/>
  <ex:composedOf rdf:resource="#CTP2"/>
  <ex:provides rdf:resource="#CON2_AddFlow"/>
</owl:Thing>
 . . .
```

Figure 4. Example of network resource instance description

```
<process:ProcessModel rdf:ID="TR1_AddFlow_ProcessModel">
  <process:hasProcess rdf:resource="#TR1_AddFlowProcess"/>
  <service:describes rdf:resource="#TR1_AddFlow"/>
</process:ProcessModel>

<process:CompositeProcess rdf:ID="TR1_AddFlowProcess">
  <rdf:type rdf:resource="#AddFlowProcess"/>
  <process:composedOf>
    <process:Sequence>
      <process:components rdf:parseType="Collection">
        <process:CompositeProcess  rdf:about="#CON1_AddFlowProcess"/>
        <process:CompositeProcess  rdf:about="#CON2_AddFlowProcess"/>
      </process:components>
    </process:Sequence>
  </process:composedOf>
</process:CompositeProcess>
 . . .
```

Figure 5. Example of COS instance description

and each of CON1 and CON2 provides the COS named "CON1_AddFlow" and "CON2_AddFlow", respectively. These COSs are instances of the AddFlow service class.

The description shown in Figure 5 represents the fact that this is a description describing the operation process model of the COS named TR1_AddFlow, and the model has the process named "TR1_AddFlowProcess". It also represents the fact that TR1_AddFlowProcess is a composite process and composed of the composite processes named "CON1_AddFlowProcess" and "CON2_AddFlowProcess", each of which is the operation process of CON1_AddFlow and CON2_AddFlow service, respectively, and TR1_AddFlowProcess is executed by executing them in sequence. The relationships among the input arguments are provided in another description.

IV. USERTRAIL COS COMPOSITION AND OPERATION

A. *UserTrail COS composition*

The COS composer composes composite COSs based on the network resource and COS descriptions. First, the COS composer extracts required network resources and combines them. Then it extracts required COSs provided by the extracted network resources, and composes composite COSs.

As regards composition of COSs for operation of UserTrails, for the particular section between two edges in the user network, UserTrails are composed by combining Connections, each of which is extracted from each Link. Then COSs for operation of the UserTrails are composed of the COSs provided by the component Connections. Some of the composite COSs provided by each of the UserTrails are composed based on the following composition rules.

- The rule for the CreateUserTrail service

  The CreateUserTrail service is a service for constructing a UserTrail. This COS will be composed if all the conditions listed below are satisfied.

  For all pairs of neighboring component Connections,

  - The lower Connection provides an AddFlow service, or it provides an AddClass service, which enables classification based on marking parameters available in the SetMarking service provided by the upper Connection.
  - The priority value ranges of the SetPriority services provided by the Connections have some common values.

- The rule for the SetMinBand service

  The SetMinBand service will be composed if all the conditions listed below are satisfied.

  - The UserTrail provides a CreateUserTrail service.
  - For all component Connections, each of them provides a SetMinBand service.

  The value range of the bandwidth is derived according to the combination of their ranges.

- The rule for the SetPriority service

  The SetPriority service will be composed if all the conditions listed below are satisfied.

  - The UserTrail provides a CreateUserTrail service.
  - For all component Connections, each of them provides a SetPriority service.

  The value range of the priority will be their common values.

$$
\begin{aligned}
&\forall tr \\
&InstanceOf(tr, UserTrail) \cap Provides(tr, CreateUserTrail) \cap \\
&(\\
&\quad \forall con \\
&\quad (InstanceOf(con, Connection) \cap Composes(con, tr)) \\
&\quad \Rightarrow Provides(con, SetMinBand) \\
&) \\
&\Rightarrow Provides(tr, SetMinBand)
\end{aligned}
$$

Figure 6. Example of composition rule

- The rule for the AddFlow service

The AddFlow service will be composed if all the conditions listed below are satisfied.

- The UserTrail provides a CreateUserTrail service.
- The uppermost Connection provides an AddFlow service.

Figure 6 shows part of an example of the composition rule for the SetMinBand service, based on the first-order logic.

B. UserTrail operation via COS

UserTrails are controlled as described below based on the COS instance descriptions.

When a COS for a UserTrail operation is called, the COS instance description of the UserTrail is interpreted. The services described in the composedOf attribute are then called in sequence. Each request for the service is taken over by the wrapper module for each piece of network equipment. Each wrapper module then converts the request for the service to proprietary control messages, and controls each piece of network equipment via proprietary operation interfaces.

For example, we assume a case where one piece of equipment is an Ethernet-Bluetooth bridge and the other is an L3-SW supporting Diffserv, and the AddFlow service provided by a UserTrail is called. Service processes in a composedOf attribute are called and the wrapper module for each piece of equipment converts the request and arguments into a proprietary control message and arguments. It sends the bridge a proprietary Bluetooth message asking it to create an ACL (Isochronous) connection. It also sends an L3-SW a proprietary message asking it to classify the flow into the EF class.

V. CONCLUSION AND FUTURE WORK

To provide end users with easy-to-use user interfaces for user network operation, we proposed a method for composing COSs over a variety of user network elements, using Web service composition techniques. We defined the user network resource and COS model, in which composite COSs are composed of other atomic or composite COSs. Composite COSs are composed based on the network resource and COS descriptions, which are provided in semantic languages for Web service composition such as OWL and OWL-S. As an example of COS composition, we studied composition of COSs for QoS control between each gateway and end points. Due to this method, practical network operation services are automatically composed in accordance with each user network configuration.

In the future, we plan to extend the UNM architecture to realize a variety of composite COSs. To realize this extended architecture, we should construct the concept system for describing the network resources and COSs, based on the semantic languages such as OWL and OWL-S. We also plan to study a method for providing end users with easy-to-use user interfaces for user network operation, and a method for composing appropriate composite COSs in accordance with users' demands.

REFERENCES

[1] "Specification of the Bluetooth System Version 1.1 ", Bluetooth SIG, February 22, 2001.

[2] N. Nishiyama, K. Nishikawa and F. Ito, "A study on user network management method realizing extensions for each user environment", *Proc. of the 2004 IEICE General Conference*, March 2004, Vol. B-14-16, pp. 601.

[3] B. Medjahed, A. Bouguettaya, A. K. Elmagarmid, "Composing Web services on the Semantic Web", *The VLDB Journal (2003) 12*, 2003, pp.333-351.

[4] N. Nishiyama, Y. Suzuki and F. Ito, "Method for user network resource management", *Technical Proc. APNOMS 2003*, October 2003, pp.212-223.

[5] "Information technology - Open Systems Interconnection - Structure of management information - Part 4: Guidelines for the definition of managed objects (GDMO)", ISO/IEC 10165-4, 1992

[6] K. McCloghrie, D. Perkins, J. Schoenwaelder, J. Case, M. Rose and S. Waldbusser, "Structure of Management Information Version 2 (SMIv2)", RFC 2578, April 1999.

[7] "OWL Web Ontology Language Overview", W3C, February 2004.

[8] "OWL-S: Semantic Markup for Web Services", OWL-S Coalition, November 2003.

[9] "OSGi Service Platform, Release 3 ", OSGi, March 2003.

[10] "Generic Network Information Model Amendment 1 ", ITU-T Recommendation M.3100 Amendment 1, 1999.

[11] "Common Elements of the Information Viewpoint for the Management of a Transport Network", ITU-T, Recommendation G.853.1, 1999.

[12] "Enterprise Viewpoint Description of Transport Network Resource Model", ITU-T, Recommendation G.852.2, 1999.

[13] "IP Network Management Information Agreement - TMF611 v0.5 - Member Draft", TeleManagement Forum, April 2002.

[14] "Multi Technology Network Management Information Agreement - TMF608 v2.1 ", TeleManagement Forum, August 2002.

[15] Y. Suzuki and I. Takahashi, "A method for access network resource management", *2003 Technical Report of IEICE*, March 2003, Vol. 102, No. 706 (TM2002-95), pp. 47-52.