

Network Service Description and Discovery for the Next Generation Internet

Qiang Duan

*Information Science and Technology Department
The Pennsylvania State University Abington College
Abington, PA 19001, USA*

qduan@psu.edu

Enyue Lu

*Mathematics and Computer Science Department
Salisbury University
Salisbury, MD, 21801, USA*

ealu@salisbury.edu

Abstract

The next generation Internet will face new challenges due to the coexisting heterogeneous networks and highly diverse networking applications. Therefore how to coordinate heterogeneous networking systems to support a wide spectrum of application requirements becomes a significant research problem. A key to solving this problem lies in effective and flexible collaborations among heterogeneous networking systems and interactions between applications and the underlying networks. Network virtualization plays a crucial role in enabling such collaborations and interactions, and the Service-Oriented Architecture (SOA) provides a promising approach to supporting network virtualization. Network service description and discovery are key technologies for applying SOA in networking, and the current service description and discovery technologies must be evolved to meet the special requirements of future Internet. In this paper, we study the problem of network service description and discovery to support network virtualization in the next generation Internet. The main contributions of this paper include a general approach to describing service capabilities of various heterogeneous networking systems, a technology to discover and select the network services that guarantee the QoS requirements of different networking applications, a general profile for specifying networking demands of various applications, a scheme of network resource allocation for QoS provisioning, and a system structure for realizing the network description, discovery, and resource allocation technologies. We also propose information update mechanisms for improving performance of the network service description and discovery system. The approach and technology developed in this paper are general and independent of network architectures and implementations; thus are applicable to the heterogeneous networking systems in the next generation Internet.

Keywords: SOA, Internet, network virtualization, service description, service discovery, QoS.

1. INTRODUCTION

Although became a global communication platform in a short period of time, the Internet has fallen victim to its own stunning success. The next generation Internet will face many challenges, which essentially come from the diversity in the networking architectures and technologies coexisting in the Internet and the wide spectrum of networking applications supported by the Internet. The numerous distributed computing applications with diverse networking requirements motivate research on alternative networking technologies and architectures for the future Internet. The newly developed networking technologies vary in almost all aspects of network functionalities, including data forwarding mechanisms, routing and signaling protocols, and control and management schemes. So far no single network architecture has demonstrated the capability of supporting all kinds of networking applications effectively and efficiently. The coexistence of various heterogeneous networking systems will be one of the essential features of the next generation Internet. Recently the networking research community started exploring “clean-slate” approaches to develop new Internet architectures, and an important aspect of the objective is to enable alternative network architectures coexist and collaborate inside future Internet. Therefore, how to coordinate heterogeneous networking systems to support the wide variety of application requirements becomes a significant research problem.

A key to solve this problem lies in flexible collaboration across heterogeneous networking systems and effective interactions between networks and applications. Recent research progresses toward this direction include new network architecture for diversifying the Internet [23], a concurrent architecture (CABO) that separates the roles of service providers and network infrastructure providers [10], a new network control plane developed in the DRAGON project [17], and the network composition mechanism for Ambient Networks [6]. Examples of current standardization efforts for supporting various applications across heterogeneous networks include the ITU-T Next Generation Network (NGN) structure [28] and the Open Service Environment (OSE) defined by the Open Mobile Alliance (OMA) [29]. Although the above researches address the heterogeneous networking problem from different aspects, they share a similarity in which the notion of *network virtualization* plays a crucial role. Essentially network virtualization is to abstract networking resources into reusable components that can be discovered, selected, and composed to meet different networking requirements. Therefore an effective mechanism for network virtualization will greatly facilitate the next generation Internet.

The *Service-Oriented Architecture* (SOA) [25] is currently gaining high attention and acceptance in IT industry, especially in the fields of Web services and Grid/Cloud Computing. In SOA, heterogeneous computational resources are virtualized into *services* that are discoverable, selectable, and composable for supporting various computing application requirements. A key feature of SOA is the loose-coupling mechanism that enables flexible and effective interactions among services and applications. Therefore applying ideas of SOA in networking will greatly facilitate network virtualization in the next generation Internet. Following the SOA principles, heterogeneous networking systems can be virtualized into *network services* which are reusable network components whose functions and access methods are described in standard-format documents called network service descriptions. The network service descriptions can be published at a service registry. When an application needs to utilize networking resources, it sends a request to a network service broker, which will discover and select the appropriate network service that meets the application requirements. Therefore network service description and discovery form the foundation of this service-oriented networking paradigm.

Although SOA has been successfully applied in the field of distributed computing, currently available technologies for service description and discovery must to be enhanced to meet the special requirements of the next generation networking. One of the key requirements is to support the network Quality of Service (QoS) required by applications. The current service description standard in SOA mainly focuses on functional information instead of service provisioning capabilities; thus limiting service discovery to be function-based instead of performance-based. Although progresses have been made toward enabling performance-based Web and Grid

services discovery, the obtained results may not be applied directly to networking systems. Therefore, new approaches to describing network QoS capability and discovering network services based on their achievable QoS must be developed.

In this paper we address the problem of service-oriented network description and discovery to meet the requirements of network virtualization in the next generation Internet. The heterogeneity of networking systems and the diversity of application requirements in future Internet make this problem challenging. The main contribution made by this paper include a general approach to describing data delivery capabilities of various heterogeneous networking systems, a technology to discover and select the network services that can guarantee the QoS required by different networking applications, a general profile for specifying the networking demands of various applications, a scheme of network resource allocation for QoS provisioning, and a system structure for realizing the developed network description, discovery, and resource allocation technologies. We also propose information update mechanisms for improving the scalability and performance of the network service description and discovery system.

The rest of this paper is organized as follows. Section 2 reviews networking for the next generation Internet and discusses the application of SOA in this area. Section 3 proposes a new approach for describing network service capabilities. Section 4 develops the technology for network service discovery and resource allocation for network service provisioning. Numerical examples are provided in Section 5 to illustrate applications of the developed technologies. In Section 6 we design a system structure for network service description and discovery, and discuss scalable mechanisms for updating network service information. Section 7 draws conclusions.

2. SERVICE-ORIENTED NETWORKING FOR THE NEXT GENERATION INTERNET

2.1 Networking for the Next Generation Internet

The next generation Internet will be featured by a wide spectrum of supported networking applications and the diverse coexisting network infrastructures. The diversity of network infrastructures referenced here includes the type of networking technologies, service capabilities and provisioning mechanisms, administrative ownership, network management and control policies, among others. The coexisting diverse networking systems must cooperate with each other in order to deliver end-to-end network services for supporting various network applications. The current Internet architecture lacks the flexibility to face this challenge, therefore developing new network architectures for enabling flexible interactions among heterogeneous networks to support various applications has become an active research area.

Recently a new network architecture was proposed for diversifying the Internet [23]. This architecture enables various meta-networks built on top of a shared substrate comprising heterogeneous networking resources. Such an architecture allows network providers to automatically deploy, configure, and operate meta-networks to meet application requirements. A similar idea was also developed in the CABO architecture (“Concurrent Architecture are Better Than One”) [10] for the future Internet. The CABO architecture decouples infrastructure providers (who manage the physical infrastructure) and network service providers (who deploy network protocols and offer end-to-end services); thus supporting multiple simultaneous network architectures on top of shared physical infrastructures. An inter-domain network control plane was developed in the DRAGON (Dynamic Resource Allocation in GMPLS Optical Networks) project [17] for enabling dynamic provisioning of networking resources for high-performance Grid and e-science applications. A crucial component in this control plane is the Network-Aware Resource Broker (NARB) that represents the local autonomous systems. A networking paradigm called Ambient Network (AN) [6] was developed to support ubiquitous provisioning of network services

over any type of network. A core feature of Ambient Network is a uniform mechanism called *network composition* for dynamic and scalable cooperation between heterogeneous networks. A service plane architecture for multi-domain network connections was also reported in [12], which introduced the notion of inter-domain network service as the result of composing a set of service elements. Some network standard organizations are also working on supporting applications across heterogeneous networks. The Open Mobile Alliance (OMA) recently developed an Open Service Environment (OSE) that delivers network services by composing a set of standard service components called service enablers [29]. The Next Generation Network (NGN) structure defined by ITU-T promotes separation between service delivery from network transport infrastructure [28].

Although the above mentioned research and standardization efforts tackle the problem of coordinating heterogeneous networking systems to support various applications from different aspects, the notion of network virtualization plays a significant role in all these proposed solutions. Through network virtualization, the underlying networking resources are de-coupled from network service provisioning. Networking resources are encapsulated into reusable components, which can be discovered, selected, and composed to meet application requirements. In the new Internet architecture proposed in [23], a meta-network is a virtual network that serves as an abstraction of a collection of heterogeneous networking resources. The CABO architecture employs network virtualization as a key mechanism to enable service providers share the underlying network infrastructures for end-to-end service delivery. In the DRAGON network control plane, a key technology for realizing NARB is network domain virtualization, which provides a mechanism to advertise a simplified view of a network domain. Network composition in Ambient Networks is based on a connectivity and resource virtualization framework that hides the differences of heterogeneous networking systems and enables applications to operate across them. The service element proposed in [12] is also a type of virtualization of networking resources, which can be discovered and composed to form inter-domain network services. The service enablers in the OMA OSE are reusable virtualizations of networking resources that provide access to network capabilities. The separation between service provisioning and network transportation in the NGN structure is also realized through network virtualization. Therefore, an effective and flexible mechanism for network virtualization will greatly facilitate the development of new networking architectures and technologies for the next generation Internet.

2.2 The Service-Oriented Architecture and Its Application in Networking

The *Service-Oriented Architecture* (SOA) is a system architecture initially developed by the IT community, especially in the areas of Web services and Grid/Cloud computing, as an effective solution to coordinating computing resources crossing heterogeneous systems to support various applications. The SOA is described as “an architecture within which all functions are defined as independent services with invocable interfaces that can be called in defined sequences to form business processes” [9]. Services in SOA are self-contained and reusable computing components that can cooperate with other services through pre-defined standard interfaces. Essentially the SOA enables virtualization of various computing resources in form of services and provides a flexible interaction mechanism among services.

The interaction mechanism in the SOA is briefly illustrated in Figure 1. A service provider publishes a machine-readable document called *service description* at a service registry. The service description gives descriptive information about the functions provided by the service and the interfaces for utilizing such functions. When a service customer, either an application or another service, needs to utilize computing resources to perform a certain function, it starts a service discovery process to locate an available service that meets its requirement. Typically a *service broker* handles service discovery for service customers by searching the service descriptions published at the registry and selecting a service that matches the criteria specified by the customer. After discovering a service, the service customer contacts the service provider and invokes the service by following the interface defined in the service description. Currently the

SOA is realized by a set of Web Services standards, including Web Service Description Language (WSDL) [35], Universal Description Discovery and Integration (UDDI) [32], Simple Object Access Protocol (SOAP) [36], which are all based on Extensible Markup Language (XML).

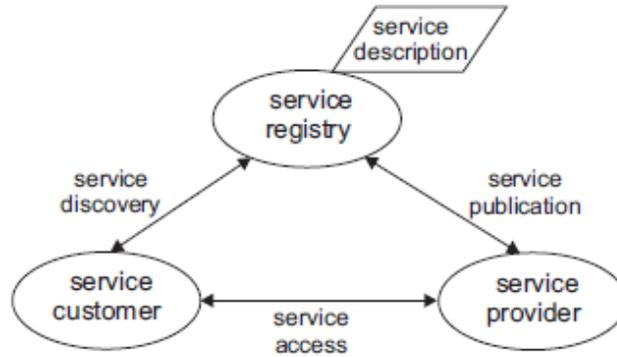


FIGURE 1: The service-oriented architecture.

A key feature of SOA is the “loose-coupling” interactions among heterogeneous systems in the architecture, including service providers, service customers, and the service broker and registry. “Loose-coupling” means entities can effectively interact with each other while keep themselves independent. It is this feature makes the SOA a very effective architecture for coordinating heterogeneous systems to support various application requirements, which is essentially the same challenge faced by the next generation Internet. Therefore, applying the SOA principles in the field of networking provides a promising approach to constructing the next generation Internet. In this paper we refer to such SOA-based networking paradigm as *Service-Oriented Networking*. In this paradigm network virtualization can be realized through encapsulating networking resources into *network services*. A network service may represent any type of networking component, could be a network domain, a collection of networks, a single physical network, or just a network node. Multiple network services can be composed into one composite inter-network service. By publishing a network service description, each network service can advertise its functions and capabilities without exposing internal implementation details.

The service-oriented networking paradigm is shown in Figure 2. In this paradigm a network service provisioning layer is deployed between networking applications and the underlying networking platform. The networking platform consists of heterogeneous networking resources encapsulated in network services. When an application needs to utilize networking resources, it sends a request to a network service broker. The network broker searches the network service descriptions published at the service registry and selects the network service that can meet the application requirement. After receiving a response from the service broker with a network selection result, the application may start accessing the networking resources provided by the selected network service.

SOA-based network virtualization gives Internet service providers and network operators the ability to view their underlying network infrastructure more as a commodity and allows infrastructure development to become more consistent. SOA-based network service provisioning also enables faster time to market as new initiatives can reuse existing services and components, thus reducing design, development, testing, and deployment time in addition to the cost and risk of undertaking such projects.

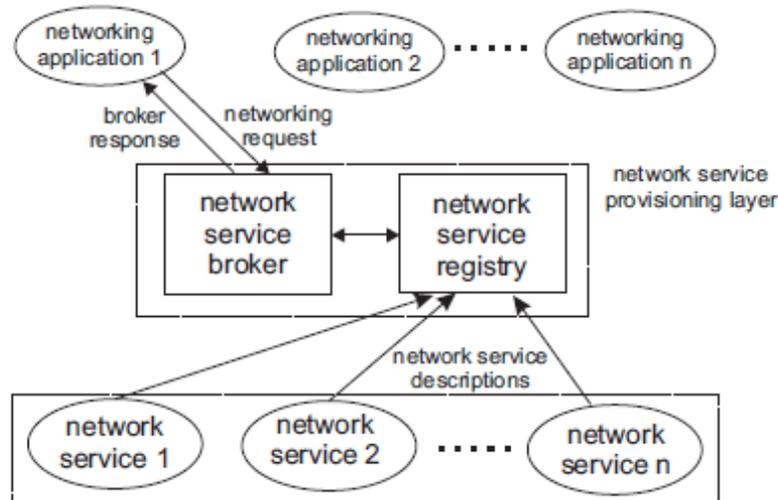


FIGURE 2: Network service description and discovery in the service-oriented networking paradigm.

2.3 Network Service Description and Discovery

Network service description and discovery are key components for adopting SOA in networking. Service-oriented network virtualization introduces new challenges to description and discovery of network services. A distinguishing feature for network services is their service provisioning capabilities, namely the capability of guaranteeing a certain level of QoS performance to an application. Therefore, service description for network virtualization should provide information about network service provisioning capability. Since most applications for the next generation Internet require high-performance networking, the key to network service discovery is selecting the appropriate network services that meet application performance requirements; that is, performance-based network service selection. The currently available service description and discovery technologies in SOA must be enhanced to meet these new requirements.

The current standards for service description and discovery are Web Service Description Language (WSDL) [35] and Universal Description, Discovery and Integration (UDDI) [32]. The WSDL specification defines an abstract interface describing the functionality of a service and a concrete interface that describes how to contact and invoke the service. The UDDI specification defines the interfaces for publishing and searching service descriptions and the data structures for organizing service description information. Currently the WSDL specification focuses on providing functional information about services and UDDI lacks effective mechanism to publish and search non-functional features such as service provisioning capability. Therefore, the service description and discovery technologies based on WSDL and UDDI are function-based instead of performance-based.

Research efforts have been made for enabling QoS-capable service description, discovery, and selection. For example, R.Al-Ali and his colleagues developed an extended UDDI registry (UDDIe) and the G-QoSM framework for supporting QoS-based Grid service discovery [1, 2]. However, the UDDIe was mainly tied with the G-QoSM framework for Grid computing and had limited support for QoS details. A QoS-capable service broker algorithm was developed to discover, select, and compose Web services that meet end-to-end QoS constraints [26, 27]. Al-Masri and Mahmoud introduced a relevancy ranking function based on QoS parameters to find the best Web service that meet client QoS preferences [3]. Technologies for QoS-aware run-time service discovery and selection were proposed in [4, 24] and a model to filter discovered services with their QoS features to maximize user satisfaction was developed in [18]. These technologies were mainly developed for Web or Grid services focusing on data processing and computing instead of data communications and networking. Therefore, the obtained results may not be applied directly to network service description and discovery.

World Wide Web Consortium (W3C) recently offered WS-Policy [37] and WS-PolicyAttachment [38] specifications for standardizing the description of non-functional characteristics of Web services. WS-Policy aims to provide a general-purpose framework and model for expressing non-functional service characteristics such as requirements, preference, and capabilities as policies. WS-PolicyAttachment defines a mechanism to associate the policy expressions with the existing WSDL standard. The WS-Agreement specification [31] developed by the Open Grid Forum (OGF) defines a protocol between service providers and users for establishing and managing service level agreements. These specifications have made significant progresses toward QoS-enable Web/Grid service description and discovery. However, modeling and describing network service provisioning capability are left opened as domain-specific issues in these specifications.

Research results on network modeling and description have also been reported. A network modeling tool called Effective Network Views was developed in [21]. The application of this tool is mainly limited to local area networks and it is not scalable to the Internet. S. Lacour and his colleagues employed the directed acyclic graph (DAG) to describe network topology and developed a scalable network description model [16]. This description model focuses on a functional view of network topology instead of service provisioning capabilities; thus lacks the information needed for performance-based network service discovery. A Network Description Language (NDL) was developed in [14] as a semantic schema for describing network topology. The NDL language serves more as a vocabulary to present network topology than an approach to modeling network service capabilities, and the reported application of NDL mainly focused on optical networks. However the next generation Internet may consist of a wide variety of networks with different implementations.

To the best of our knowledge little work has been reported with regard to network capability description and performance-based network discovery technologies that are applicable to the heterogeneous networks in the next generation Internet. The research presented in this article addresses this problem by developing a general approach for describing network service capabilities and a technology for discovering network services that meet the QoS performance required by different networking applications.

3. NETWORK SERVICE DESCRIPTION

In this section, we will develop a new approach to describing network service capabilities for supporting the service-oriented network virtualization. The main challenge to network service description lies in the heterogeneity of the networking systems that will be coexisting in the Internet. Therefore, the network service description approach must be (a) general so as to be applicable to various network implementations and (b) flexible so as to support composing multiple heterogeneous networks into one cross-domain network service. A key for network service description is to provide sufficient amount of information about the data delivery capability of a network without exposing its implementation details. In general, network data delivery capability includes two aspects: the connectivity of the network service, which can be described by enumerating the pairs of sources and destinations between which the network transports data; and the capability of data transportation between each pair of source-destination

In order to provide a formal description for network service capabilities, we define the Capability Matrix **C**. Given a network service *S* with *m* ingress ports and *n* egress ports, the capability matrix **C** is an $m \times n$ matrix

$$\mathbf{C} = \begin{pmatrix} \mathbf{C}_{1,1} & \mathbf{C}_{1,2} & \cdots & \mathbf{C}_{1,n} \\ \mathbf{C}_{2,1} & \mathbf{C}_{2,2} & \cdots & \mathbf{C}_{2,n} \\ \cdots & \cdots & \cdots & \cdots \\ \mathbf{C}_{m,1} & \mathbf{C}_{m,2} & \cdots & \mathbf{C}_{m,n} \end{pmatrix} \quad (1)$$

where \mathbf{C}_{ij} is defined as

$$\mathbf{C}_{i,j} = \begin{cases} 0 & \text{if no route exists from } i \text{ to } j \\ \mathbf{Q}_{i,j} & \text{if a route } R_{i,j} \text{ exists from } i \text{ to } j \end{cases} \quad (2)$$

and $\mathbf{Q}_{i,j}$ is called the QoS descriptor for the route $R_{i,j}$, which will be further developed in the rest of this section.

According to the definitions given in (1) and (2), the capability matrix element $\mathbf{C}_{ij} = 0$ if the network service cannot reach the egress j from the ingress i . That is, all non-zero elements in the matrix \mathbf{C} describes the connectivity of the network services. If the network service provides a route from i to j , then the transportation capability of this route is described by the descriptor $\mathbf{Q}_{i,j}$.

We adopt the notion of service curve from network calculus theory [7] to design a general QoS descriptor that is applicable to various network implementations. The network calculus theory has evolved to an effective tool for network modeling and analysis. The service curve is defined as follows in network calculus. Let $T^{in}(t)$ and $T^{out}(t)$ respectively be the accumulated amount of traffic of a flow that arrives at and departs from a server by time t . Given a non-negative, non-decreasing function, $S(\cdot)$, where $S(0) = 0$, we say that the server guarantees a service curve $S(\cdot)$ for the flow, if for any $t \geq 0$ in the busy period of the server,

$$T^{out}(t) \geq T^{in}(t) \otimes S(t) \quad (3)$$

where \otimes denotes the min-plus convolution operation defined in network calculus as $h(t) \otimes x(t) = \inf_{s:0 \leq s \leq t} \{h(t-s) + x(s)\}$. Essentially a service curve gives the minimum amount of service offered by the server to a client in an arbitrary time interval within a busy period. Therefore a service curve describes the lower bound of the service provisioning capability offered to a client.

A typical server model for networking systems is the Latency-Rate (LR) server [22], which guarantees each flow a service curve $\beta_{r,\theta}(t) = r(t - \theta)$, where θ and r are respectively called the latency and service rate for the flow. LR server is particularly interesting to us because many packet schedulers such as Weighted Fair Queuing (WFQ) and Weighted Round-Robin (WRR), which are widely deployed in practical networking equipments, belong to this server category.

In our service description approach, we adopt the service curve guaranteed by the route $R_{i,j}$ as the QoS descriptor $\mathbf{Q}_{i,j}$ in the matrix \mathbf{C} . Since a service curve is a general data structure that is independent with network implementations, it is flexible enough to describe various networking systems. In a network where a route $R_{i,j}$ can be modeled by a LR server with a service curve $r_{i,j}(t - \theta_{i,j})$, the matrix element $\mathbf{C}_{i,j}$ can be represented by a data structure $[r_{i,j}, \theta_{i,j}]$. Currently there are various mechanisms available for measuring and managing network state information, for example the technologies reported in [13,15, 20], which could be used to obtain the data for constructing service curves and building the matrix \mathbf{C} . The methods of collecting network state information are implementation dependent and may vary in different networks, but the matrix \mathbf{C} provides all network services with a general and standard approach to describing their service provisioning capabilities.

An end-to-end Internet connection typical crosses multiple networks, each of which can be virtualized as a network service. Therefore, how to compose the QoS capabilities of a set of heterogeneous links into one descriptor for the end-to-end connection is an important and challenging problem. The service curve-based description approach supports QoS descriptor composition. Known from network calculus, the service curve $S(t)$ guaranteed by a series of

tandem servers G_1, G_2, \dots, G_n , which respectively guarantees the service curves $S_1(t), S_2(t), \dots, S_n(t)$ to a flow, can be obtained through the convolution of all the service curves; that is,

$$S(t) = S_1(t) \otimes S_2(t) \cdots \otimes S_n(t) \quad (4)$$

Therefore, the QoS descriptor of the end-to-end route can be obtained from the convolution of the QoS descriptors of the links provided by all single network services.

Since typical networking systems can be modeled as LR servers, we are particularly interested in composing LR servers. Suppose each network server $S_i, i = 1, 2, \dots, n$, guarantees a service curve $\beta_{r_i, \theta_i} = r_i(t - \theta_i)$, it can be proved that the convolution of these service curves is

$$\beta_{r_1, \theta_1}(t) \otimes \cdots \otimes \beta_{r_n, \theta_n}(t) = \beta_{r, \theta_\Sigma}(t) \quad (5)$$

where

$$r = \min\{r_1, r_2, \dots, r_n\} \quad \text{and} \quad \theta_\Sigma = \sum_{i=1}^n \theta_i \quad (6)$$

Equations (5) and (6) imply that if each link on an end-to-end network route can be described by a latency-rate server, then the end-to-end route also guarantees a latency-rate service curve, whose latency parameter is equal to the summation of all link latency parameters and the service rate parameter is limited by the link with the least service rate.

In this section we developed a general description for network service capabilities. Due to the network calculus technique employed in our development, this description approach is independent with network implementations; thus applicable to various heterogeneous networking systems in the Internet. This approach can also easily support capability description for composite network services.

4. NETWORK SERVICE DISCOVERY

In this section we develop a new technology for performance-based discovery of network services, which enables the network service broker to discover network services that guarantee QoS performance required by networking applications. This newly developed technology focus on network service selection while other components of the discovery procedure, including publishing service descriptions, searching the registry for available services, negotiating service level agreement, and binding the selected service with the application, can be implemented based on current SOA service discovery standards.

4.1 Performance Prediction for Network Service Discovery

Three aspects of information are needed by a network service broker for performance-based network service discovery for an application: (a) the provisioning capabilities of available network services; (b) the performance requirement of the application; and (c) the characteristic of network traffic generated by the application. The information (a) can be obtained from the capability matrix **C** published by the network service provider. The other two aspects of information (b) and (c), which specify the demand of a networking application, should be provided to the service broker by the application as part of its request.

Due to the large number of networking applications with various requirements, it is very important to have a common approach for describing networking demands. In this paper we define a *Demand Profile* **P** (**d**, **L**, **a**) as a general specification of application requirements. This profile consists of three elements: an address set **d**; a traffic load descriptor **L**; and a performance requirement set **a**. The address set **d** specifies the addresses of the source and destination of data transportation required by the application. If the application only needs unicast (point to point) data delivery, the address set will consist of a pair of network addresses. If multicast (point to multi-points or multi-points to multi-points) communication is required, the set **d** may include multiple address pairs. The set **a** consists of the performance parameters required by the

application. Different parameters may be included in \mathbf{a} for different applications, but the minimum bandwidth b_{req} and the maximum delay d_{req} for data transportation are typical requirements; i.e. typically $\mathbf{a} = \{b_{req}, d_{req}\}$. The descriptor \mathbf{L} is used to characterize the network traffic that the application will load on a network service.

In order to describe application traffic loads in a general form, we employ the *arrival curve* as the traffic descriptor \mathbf{L} . Arrival curve is another important concept in the network calculus theory. Let $T^{in}(t)$ denote the accumulated amount of traffic generated from an application by time t . Given a non-decreasing, non-negative function, $A(\bullet)$, the application is said to have an *arrival curve* $A(\bullet)$ if for any nonnegative t and s

$$T^{in}(t) - T^{in}(s) \leq A(t - s). \quad (7)$$

Essentially the arrival curve of an application gives an upper bound for the amount of traffic that the application loads on a network service.

Currently most QoS-capable networks apply traffic regulation mechanisms at network boundaries to shape arrival traffic from applications. The traffic regulators most commonly used in practice are leaky buckets. A traffic flow constrained by a leaky bucket has an arrival curve

$$A(t) = \min\{Pt, \sigma + \rho t\},$$

where P , ρ , and σ are respectively the peak rate, the sustained rate, and the maximal burst size of this flow.

Now we develop a technique to predict the performance that can be guaranteed by a network service to a networking request. Among various performance requirements, in this paper we focus on the minimum bandwidth and the maximum delay for data transportation, which are important performances required by most high-performance networking applications. Network calculus provides us with an effective approach for analyzing the minimum bandwidth and maximum delay performances guaranteed by a network service. A service curve itself is a description of the minimum service capacity offered by a network, which essentially gives the minimum bandwidth guaranteed by the network to an application. Therefore, given the QoS descriptor for a route R_{ij} , which is described by a service curve $S_{ij}(t)$, the minimum bandwidth guaranteed by this route can be determined as

$$b_{\min} = \lim_{t \rightarrow \infty} [S_{ij}(t) / t]. \quad (8)$$

Suppose the traffic load of a networking request is described by an arrival curve $A(t)$, then the maximum delay d_{\max} guaranteed by the network to this request can be determined as,

$$d_{\max} = \max_{t \geq 0} \{\min\{\delta : \delta \geq 0, A(t) \leq S(t + \delta)\}\}. \quad (9)$$

Since the LR server is a typical network server model and the leaky bucket is a typical traffic regulator, we specifically give the performance analysis for a network route modeled by a LR server under traffic load constrained by a leaky bucket regulator. Suppose the load descriptor of an application is $A(t) = \min\{Pt, \sigma + \rho t\}$, and the QoS descriptor of the route provided by a network to this application is $Q = [r, \theta]$, then the minimum bandwidth that can be guaranteed by the route is

$$b_{\min} = \lim_{t \rightarrow \infty} \frac{r(t - \theta)}{t} = \lim_{t \rightarrow \infty} \left(r - \frac{r\theta}{t} \right) = r. \quad (10)$$

By following (9) we can get that the maximum delay guaranteed to the application is

$$d_{\max} = \theta + \left(\frac{p - r}{p - \rho} \right) \frac{\sigma}{r} \text{ for } r \geq \rho. \quad (11)$$

After determining the achievable performance of a network service S for an networking request R , the network broker compares the predicted performance with the requirement given in the set \mathbf{a} of the demand profile \mathbf{P} to decide if S can be selected for R . Networking requests can be classified into three categories according to their networking performance requirements: (a) with only bandwidth requirement; i.e. $\mathbf{a} = b_{req}$; (b) with only delay requirement; i.e. $\mathbf{a} = d_{req}$; and (c) with both

bandwidth and delay requirements; i.e. $\mathbf{a}=\{b_{req}, d_{req}\}$. If R belongs to category (a), then S can be selected only when $b_{min} \geq b_{req}$. If R belongs to category (b), then S can be selected only when $d_{max} \leq d_{req}$. If R belongs to category (c), then S can be selected only when $b_{min} \geq b_{req}$ and $d_{max} \leq d_{req}$. If there are multiple network services meet the performance requirements, selection among them may be based on other criteria such as service cost or load balance.

4.2 Resource Allocation for Network QoS Provisioning

The selected network service should allocate sufficient amount of networking resources in order to actually guarantee network QoS provisioning. The main resource in a network service is bandwidth, therefore in this subsection we give a discussion on bandwidth allocation for QoS provisioning in network services.

Equation (11) shows that given the traffic parameters (P, ρ, σ) of a networking request, the achievable delay upper bound d_{max} is a function of the available bandwidth r . This implies that the required delay performance can be guaranteed by allocating sufficient amount of bandwidth. Equation (11) also shows that the minimum possible delay $D_{min} = \theta$ when $r = P$; that is when the allocated bandwidth is equal to the traffic peak rate. Although achieving optimal delay performance, allocating bandwidth according to the peak rate causes low resource utilization, especially for applications with fluctuating load. We can also see from (11) that d_{max} is upper bounded only if $r \leq \rho$; that is, the allocated bandwidth should be at least the sustain rate ρ of the traffic load in order to achieve any delay performance guarantee. Therefore, a reasonable bandwidth allocation scheme is to determine the minimum bandwidth r_a ($\rho \leq r_a \leq P$) that is sufficient to guarantee the required delay d_{req} given by a networking request.

Analysis on (11) shows that d_{max} is a decreasing function of r that achieves the maximum value $D_{max} = \theta + \sigma / \rho$ when $r = \rho$. This implies that if $r > \rho$, the application will be guaranteed a delay upper bound that is less than D_{max} . Given the delay requirement d_{req} , the network service must guarantee a delay upper bound no greater than d_{req} ; that is,

$$d_{max} = \theta + \left(\frac{P-r}{P-\rho} \right) \frac{\sigma}{r} \leq d_{req}. \quad (12)$$

Therefore, the minimum bandwidth that must be allocated for meeting (ref{delay2}) is

$$r_a = \frac{P\sigma}{(P-\rho)(d_{req}-\theta) + \sigma}. \quad (13)$$

Equation (13) implies that r_a is always less than the peak rate P . As traffic becomes more smooth; that is the sustain rate ρ gets closer to the peak rate P , the required bandwidth r_a approaches P .

In summary, the bandwidth allocation for guaranteeing a delay requirement d_{req} can be determined as

$$r_a = \begin{cases} \rho & d_{req} \geq D_{max} \\ \frac{P\sigma}{(P-\rho)(d_{req}-\theta) + \sigma} & D_{min} \leq d_{req} \leq D_{max} \\ N/A & d_{req} < D_{min} \end{cases} \quad (14)$$

where $D_{max} = \theta + \sigma / \rho$ and $D_{min} = \theta$.

If the networking request has both delay and bandwidth requirements; i.e. $\mathbf{a}=\{b_{req}, d_{req}\}$, then the minimum amount of bandwidth that must be allocated for this request will be

$$b_{min} = \max\{r_a, b_{req}\}. \quad (15)$$

Specifically we analyze bandwidth allocation for network services that can be modeled by the weighted fair queuing (WFQ) server [19], which is widely deployed in practical networking systems. It is known from [22] that for a WFQ the latency parameter for a traffic flow is $\theta = L(1/R$

+ 1/r), where L is the maximum packet length of this flow, R is the total transmission capacity of the network route, and r is the bandwidth available to the flow on this route. For a networking application that generates a leaky bucket constrained flow with traffic parameters (P, ρ, σ) , if the QoS descriptor of the route provided by a network service for the application is $S=[r, \theta]$, then we can predict that the maximum delay guaranteed to the application is

$$d_{\max} = L\left(\frac{1}{R} + \frac{1}{r}\right) + \left(\frac{P-r}{P-\rho}\right)\frac{\sigma}{r}. \quad (16)$$

Given a delay requirement d_{req} , $D_{\min} \leq d_{req} \leq D_{\max}$, the bandwidth allocation requirement can be determined as

$$r_a = \frac{P\sigma + L(P - \rho)}{(P - \rho)(d_{req} - L/R) + \sigma}. \quad (17)$$

5. NUMERICAL EXAMPLES

In this section, we use numerical examples to illustrate the applications of the performance prediction and bandwidth allocation techniques developed for network service discovery. We considered two networking applications utilizing a network service in our examples. The application A_1 uses the network to deliver a stream of video packets and loads the network with a flow f_1 . The application A_2 transmits a flow of audio packets f_2 through the network. Both A_1 and A_2 require a small maximum packet transmission delay. We adopt the traffic parameters given in [11] for f_1 ; that is, the peak rate $P=5.3$ Mb/s, the sustained rate $\rho =1.5$ Mb/s, and the maximum burst size $\sigma =140$ kbits. The traffic parameters for f_2 are given in [8]; that is, the peak rate $P=3.2$ Mb/s, the sustained rate $\rho =1.1$ Mb/s, and the maximum burst size $\sigma =300$ kbits. We assume that the maximum transmission unit (MTU) of the network is $L=1$ K bytes, and the total link capacity of the route is $R=1$ Gb/s.

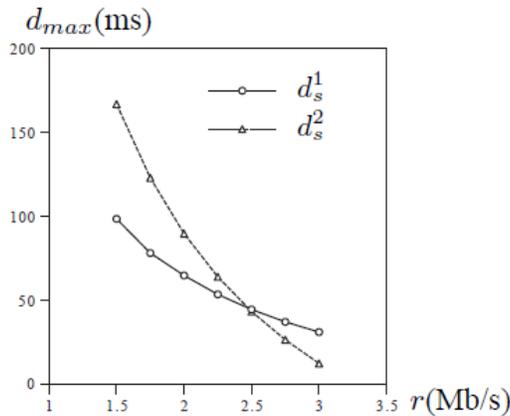


FIGURE 3: Delay performance prediction for a single network service

We first analyzed the maximum packet delay d_{max} that can be guaranteed for the flows f_1 and f_2 by a single network service with various amounts of available bandwidth r on the assigned network routes. The results are plotted in Figure 3, where d_s^1 and d_s^2 denote the maximum delay for f_1 and f_2 respectively. From this figure we can see that both d_s^1 and d_s^2 decrease when the available bandwidth r increases. This means that the more bandwidth is available to an application, the tighter is the delay upper bound guaranteed to the application. Comparison between the curves of d_s^1 and d_s^2 shows that although both of them are decreasing functions of r , d_s^2 drops faster than d_s^1 , which means that the same amount of bandwidth increment can make more significant improvement in delay performance for f_2 than what it does for f_1 . This observation implies that the QoS performance guaranteed by a network service to an application is associated with application traffic characteristics as well as the QoS capability of the network.

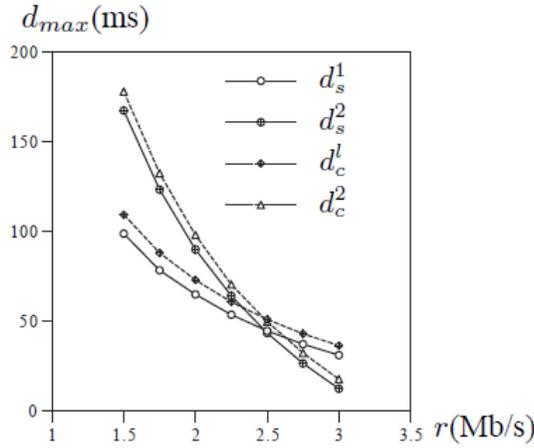


FIGURE 4: Delay performance prediction for a composite network service

We also predicted delay performance for A_1 and A_2 in a composite network service consisting of three network domains. To simplify the analysis, we assume that the three domains provide an identical QoS descriptor for each application. We calculated the maximum end-to-end packet delay for the traffic flows f_1 and f_2 that pass through the three domains, which are denoted by d_c^1 and d_c^2 respectively. We plotted the results of d_s^1 and d_s^2 with data for d_c^1 and d_c^2 in Figure 4. From this figure we can see that d_c^1 and d_c^2 are also decreasing function of available bandwidth r . This figure also shows that for any available bandwidth r , $d_s^1 < d_c^1$ and $d_s^2 < d_c^2$. This implies that the maximum packet delay guaranteed to each application by the composite network service is always greater than what is guaranteed by a single network service in the composite service. An interesting observation in Figure 4 is that for each application, the delay upper bound achieved by the composite network service is only slightly greater than the maximum delay guaranteed by a single service, which is much less than the summation of the maximum delays of all domains passed by the traffic flow.

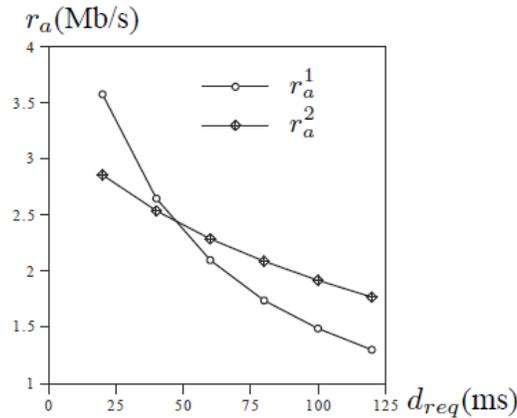


FIGURE 5: Bandwidth allocation for delay performance guarantee

We also analyzed the bandwidth allocation requirements in a network service to guarantee a set of delay upper-bounds for the two applications. The results are plotted in Figure 5, where r_a^1 and r_a^2 are respectively the required amounts of bandwidth for applications A_1 and A_2 to achieve the required delay upper-bound d_{req} . From this figure we can see that both r_a^1 and r_a^2 are decreasing functions of d_{req} . This means that the tighter the required delay upper-bound is, the more bandwidth must be allocated in the network to guarantee it. Figure 5 also shows that r_a^1 decreases with the increment of d_{req} faster than r_a^2 does. This implies that in the same network service, applications with different traffic load characteristics need different amounts bandwidth to

achieve the same level of delay performance. This also justifies the necessary of having a traffic load descriptor L as part of the networking demand profile P for performance-based network service discovery.

6. NETWORK SERVICE DESCRIPTION AND DISCOVERY SYSTEM

In this section, we discuss a system for realizing the network service description and discovery technologies developed in previous sections. Figure 6 shows the structure of this system, which consists of a Network Service Broker (NSB), a Network Service Registry (NSR), multiple Network Service Providers (NSPs), and a Network Service Consumer (NSC). A network service provider could be single networking system, a network domain with multiple networking systems, or a collection of network domains that belong to an Internet Service Provider (ISP). The network service consumer could be user equipment, or a networking application, or maybe a network service provider that needs to access networking resources managed by other network service providers.

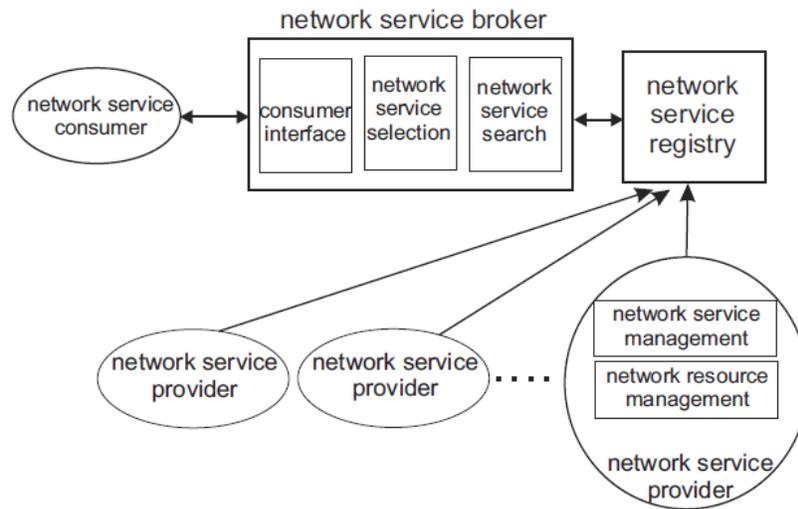


FIGURE 6: A system structure for network service description and discovery.

Each network service provider is required to support two functional modules for realizing the service-oriented networking paradigm: a Network Service Management (NSM) module and a Network Resource Management (NRM) module. The NSM module registers the network service at the broker, collects network state information from the NRM module to compile a network service description (including the capability matrix \mathbf{C}), and publishes the description at the service registry. The NSM module is also responsible for updating the network service description at the service registry when network information changes.

The network service broker consists of three modules: Consumer Interface (CI), Network Service Discovery (NSD), and Network Service Selection (NSS). When a network service consumer needs to discover a service provider for accessing networking resources, it submits a networking request to the service broker through the CI interface. This request includes a networking demand profile $P[\mathbf{d}, L, \mathbf{a}]$. On receiving the networking request and the demand profile, the discovery module (NSD) of the broker accesses the service registry to find all available network services that provide a network route between the source and destination specified in \mathbf{d} of the demand profile. These network services are selected as candidate services and their capability matrixes will be further analyzed by the broker. For each candidate service, the selection module (NSS) uses the QoS descriptor from the matrix \mathbf{C} and the load descriptor L in the demand profile P to evaluate the achievable QoS performance that the service can offer to the consumer. Then the NSS module compares the predicted performance with the requirements given in the set \mathbf{a} of the profile P and selects the service that meets all performance requirements given by the

consumer. If multiple candidate network services meet the requirements, selection among them is based on other criteria such as service cost or load balance. After selecting a network service, the network service broker will send a reply message through the CI interface to the service consumer to notify the network selection result. Then the consumer may contact the selected network service provider to negotiate a Service Level Agreement (SLA). The NRM module in the network service provider may need to allocate a certain amount of resources for QoS provisioning. After resource reservation, the NSM module of the service provider will accordingly update the capability matrix **C** and republish the latest service description at the service registry. The interactions among service consumer, broker, registry, and provider for network service description and discovery is shown in Figure 7.

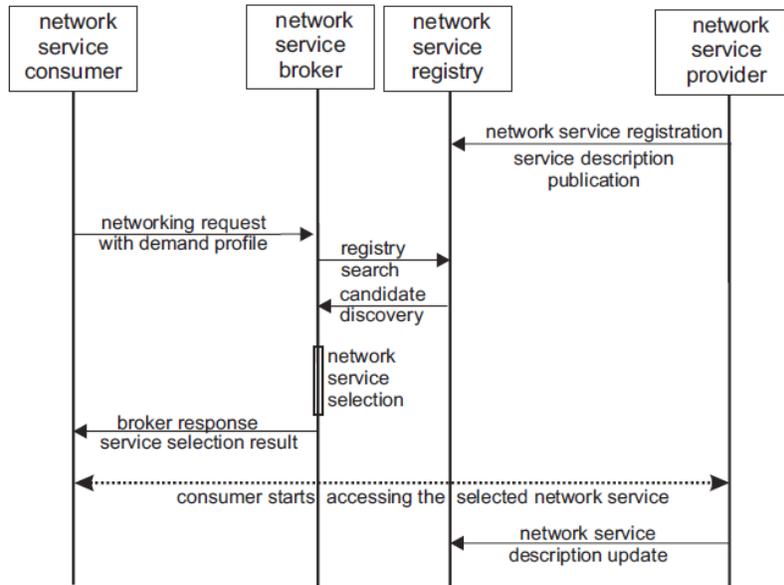


FIGURE 7: Interaction procedure for network service description and discovery.

In this paper we focus on capability description and QoS-based selection for network services. Other technologies for implementing the network description and discovery system, including publishing service descriptions, maintaining service descriptions at the registry, and searching the registry for candidate networks can be implemented based the current SOA service description and discovery standards [32, 34, 35]. The negotiation and establishment of a service level agreement between the network service consumer and provider can be implemented based on the WS-Agreement specification [31].

In a large scale dynamic networking environment such as the Internet, the states and capability information of various networking systems change frequently. Therefore keeping the latest network description information at the network service registry is significant for discovering and selecting the appropriate network services. However, republishing the entire service description, including the capability matrix **C**, whenever network state changes cause a large amount of communications and management overhead between service providers and the registry, and make the service registry a performance bottleneck. In order to solve this problem, an event-driven subscription-notification mechanism can be applied to reduce the overhead caused by frequent description update; thus improving the overall performance of the service discovery system.

Event-driven processing and notification introduces a notification pattern for SOA implementations. In this pattern an information producer sends one-way notification messages to one or more interested receivers. The message typically carries information about an event that has occurred. The message receivers must register to the producer before receiving the notifications. The OASIS Web Service Notification [33] is a family of related specifications that

define a standard approach to notification using a topic-based subscription-publication pattern. Following this pattern, the network service registry can subscribe to a network service provider and specify a set of network states as subscription topics. Then, the service registry will receive a notification message from the service provider whenever a network state subscribed as a topic changes. A threshold can also be set to each subscription topic so that update notification is only triggered when the amount of change in that topic is greater than the threshold. In this way, the network service registry can obtain the latest network state and capability information for supporting real-time network discovery and selection. Since an update only happens when a network state changes more than a pre-specified threshold and the notification message contains only the changed states instead of the entire description document, this updating mechanism can greatly reduce communications and management overhead.

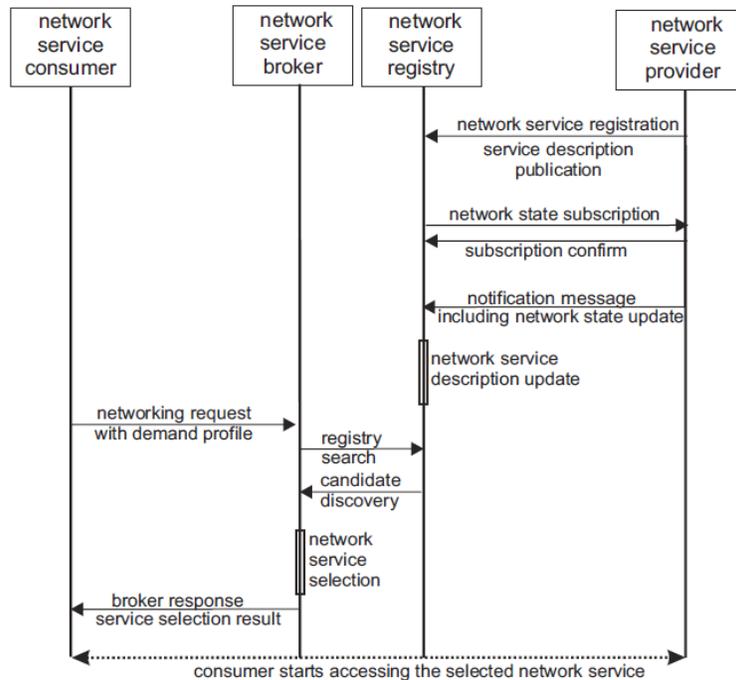


FIGURE 8: Event driven network service description update.

The interaction procedure for network service description and discovery with event-driven information update is shown in Figure 8. When a network service provider first time publishes its network service description at the service registry, the registry will also subscribe itself to the service provider to receive notification for network state change. The registry can specify what network states are subscription topics. After this registration-subscription procedure completed, the network service registry will be notified whenever the specified subscription topics change in the network service. Then the description for this network service will be updated at the registry accordingly.

The performance of the network service description and discovery system could be further improved by reducing the information updating overhead through a partial publication technology. This technology allows network service providers publish only part of their service descriptions that are relatively static at the service registry. This part of description could include information such as the network types (e.g., connectionless IP network or circuit switching telecommunication network, cellular mobile network or WiFi WLAN), network service operators (e.g., AT&T or Verizon), and also the connectivity parameters of the capability matrix **C**, which are relatively stable for typical networks. If a network service is selected as a candidate, then the service selection module of the broker will contact the NSM module in the provider of this service to retrieve additional information needed for decision making on network service selection, for example retrieving the QoS descriptor of the matrix **C** for performance evaluation. In this way the

dynamic network state information such as the QoS capability descriptors of each network service is maintained within the service provider itself instead of at the service registry; thus can be updated in real time without generating communication overhead between networks and the service registry. The partial publication technology can be used together with the event-driven update mechanism. Update on the published part of description can be implemented by the notifications from the network service provider. Since only the relatively static part of description is published at the registry, the update frequency and the messaging load will be further reduced. Figure 9 shows the interactions for network service discovery and selection with partial description publication.

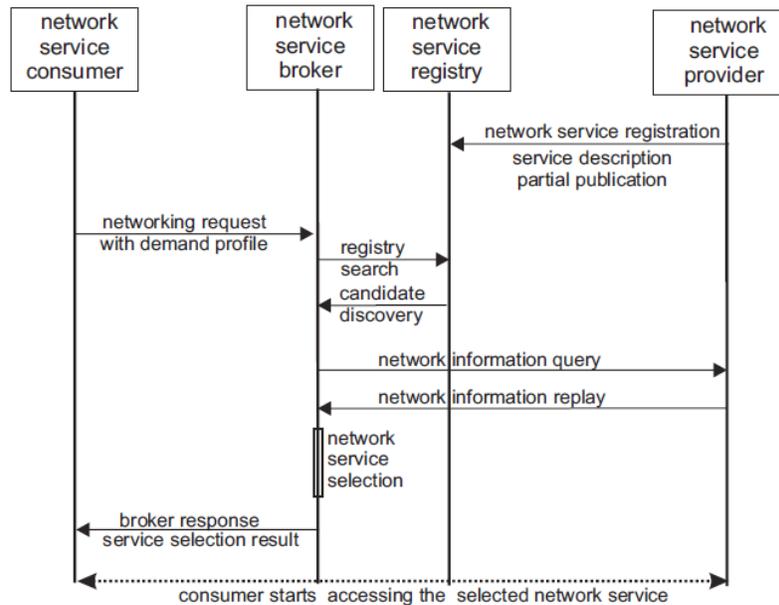


FIGURE 9: Partial publication of network service description.

7. CONCLUSIONS

The next generation Internet will be highly diversified in both underlying networking technologies and supported networking applications. Therefore coordination among heterogeneous networks to support the wide spectrum of application requirements is significant for building the next generation Internet. Although recently various research projects started addressing this challenge from different aspects, the notion of network virtualization plays a crucial role in all the efforts. The Service-Oriented Architecture, which has been widely applied in Web services and Grid/Cloud computing areas, provides a promising approach to supporting network virtualization. Network service capability description and performance-based network service selection are keys to enabling the application of SOA in the next generation Internet.

In this paper we developed a general approach to describing service capabilities of heterogeneous networking systems and a technology for selecting the network services that can guarantee application performance requirements. The core component of the network description approach is a capability matrix that describes both connectivity and QoS capability of a network service. The key of the network selection technology is to predict the achievable performance for a network service and check it against the performance requirement specified by the application. In order to allow various applications to specifying their networking requests, this paper gives a general demand profile that includes a traffic load descriptor and a performance requirement set. We also developed a scheme for allocation resources in network services for QoS provisioning. A system structure for realizing the network service description and discovery technologies is also described in this paper. In a large scale dynamic networking environment such as the Internet, keeping network service descriptions up-to-date is challenging and may cause significant

communication and management load. We proposed an event-driven information update mechanism and a partial description publishing technology that may significantly improve the performance of the network service description and discovery system. Our analysis methods and the developed technologies for network service description, discovery, resource allocation, and information update are general and independent of network implementations; thus are applicable to heterogeneous networking systems in the next generation Internet. The service description and performance analysis techniques are also flexible to support composite network services that comprise multiple networking systems.

8. REFERENCES

1. R. J. Al-Ali, O. M. Rana and D. W. Walker. "G-QoS: Grid service discovery using QoS properties". *J. of Computing and Informatics*, 21(4):1--15, 2002
2. R. J. Al-Ali, A. ShaikhAli, O. M. Rana and D. W. Walker. "Supporting QoS-based discovery in service-oriented grids". In *Proceedings of the 17th Intl. Parallel and Distributed Processing symposium*, 2003
3. E. Al-Marsri and Q. H. Mahmoud. "QoS-based discovery and ranking of Web services". In *Proceedings of the 16th IEEE International Conference on Computer Communications and Networks*, August 2007
4. E. Ambrosi, M. Bianchi, C. Gaibisso, G. Gambosi and F. Lombardi. "A system for predicting the run-time behavior of Web Services". In *Proceedings of the 2005 International Conference on Services Systems and Services Management*, June 2005
5. T. Anderson, L. Peterson, S. Shenker and J. Turner. "Overcoming the Internet impasses through virtualization". *IEEE Computer Magazine*, 38(4):34--41, 2005
6. F. Belqasmi, R. Glitho and R. Dssouli. "Ambient Network Composition". *IEEE Network Magazine*, 22(4):6--12, 2008
7. J. L. Boudec and P. Thiran. "Network calculus: a theory of deterministic queueing systems for the Internet", Springer Verlag LNCS 2050, June 2001
8. M. Butto, E. Cavallero and A. Tonietti. "Effectiveness of the leaky bucket policing mechanisms in ATM networks". *IEEE J. Select. Areas Commun.*, 9(4):335--342, 1991
9. K. Channabasavaiah, K. Holley and E. Tuggle. "Migrating to a Service-Oriented Architecture". *IBM DeveloperWorks*, Dec. 2003
10. N. Feamster, L. Gao and J. Rexford. "How to lease the Internet in your spare time". *ACM SIGCOMM Computer Communications Review*, 37(1):61--64, 2007
11. P. Fizek and M. Reisslein. "MPEG-4 and H.263 video traces for network performance evaluation". *IEEE Network Magazine*, 15(6):40--54, 2001
12. R. Douville, J.-L. Le Roux, J.-L. Rougier and S. Secci. "A service plane over the PCE architecture for automatic multidomain connection-oriented services". *IEEE Communications Magazine*, 46(6):90--102, 2008.
13. T. S. Eugene Ng and H. Zhang. "Predicting Internet network distance with coordinates-based approaches". In *Proceedings of IEEE INFOCOM'02*, New York, June 2002

14. J. Ham, P. Grosso, R. Pol, A. Toonk and C. de Laat. "Using the network description language in optical networks". In Proceedings of the 10th IFIP/IEEE Intl. Symposium on Integrated Network Management, May 2007
15. A. Kind, X. Dimitropoulos, S. Denazis and B. Claise. "Advanced network monitoring brings life to the awareness plane". IEEE Communications Magazine, 46(10):140--146, 2008
16. S. Lacour, C. Perez and T. Priola. "Network topology description model for Grid application deployment". In Proceedings of the 5th IEEE/ACM Intl. Workshop on Grid Computing, Nov. 2004
17. T. Lehman, J. Sobieski and B. Jabbari. "DRAGON: A Framework for Service Provisioning in Heterogeneous Grid Networks". IEEE Communications Magazine, 44(3):84--90, 2006
18. M. Li, B. Yu, O. Rana and Z. Wang. "Grid service discovery with rough sets". IEEE Transactions on Knowledge and Data Engineering, 20(6):851--862, 2008
19. A. K. Parekh and R. G. Gallager. "A generalized processor sharing approach to flow control in integrated services networks: the single-node case". IEEE/ACM Trans. Networking, 1(3):344--357, 1993
20. R. Prasad, M. Murray, C. Dovrolis and K. Claffy. "Bandwidth estimation: metrics, measurement techniques, and tools". IEEE Network Magazine, 17(6):27--35, 2003
21. G. Shao and F. Berman. "Using effective network views to promote distributed application performance". In Proceedings of the 1999 Intl. Conf. on Parallel and Distributed Processing Techniques and Applications, June 1999
22. D. Stiliadis and A. Varma. "Latency-rate servers: a general model for analysis of traffic scheduling algorithms". IEEE/ACM Trans. Networking, 6(5):611--624, 1998
23. J. Turner and D. E. Taylor. "Diversifying the Internet". In Proceedings of IEEE Globecom 2005, Nov. 2005
24. C. Wan, C. Ullrich, L. Chen, R. Huang, J. Luo and A. Shi. "On solving QoS-aware service selection problem with service composition". In Proceedings of the 7th International Conference on Grid and Cooperative Computing, Oct. 2008
25. S. Weerawarana, F. Curbera, F. Leymann, T. Storey and D. F. Ferguson. "Web Services Platform Architecture". Prentice Hall, 2005.
26. T. Yu and K-J. Lin. "The design of QoS broker algorithms for QoS-capable Web Services". International Journal of Web Services Research, 1(4):10--16, 2004
27. T. Yu and K-J. Lin. "Service selection algorithm for Web services with end-to-end QoS constraints". Journal of Information Systems and E-Business Management, 3(2):20--30, 2005
28. ITU-T, "Functional Requirements and Architecture of the NGN Release 1," Recommendation Y.2012, Sept. 2006
29. Open Mobile Alliance, "OMA Enabler Releases and Specifications - OMA Service Environment Architecture Document," Nov. 2007
30. Open Grid Forum (OGF) OGSA-WG, "The Open Grid Service Architecture, version 1.0," Jan. 2005

31. Open Grid Forum (OGF) "Web Services Agreement Specification (WS-Agreement)," March 2007, available online: <http://www.ogf.org/documents/GFD.107.pdf>
32. Organization for the Advancement of Structured Information Standards (OASIS), "Universal Description, Discovery and Integration Version 3.0.2," Feb. 2005, available online: <http://www.oasis-open.org/specs/>
33. Organization for the Advancement of Structured Information Standards (OASIS), "Web Services Base Notification (WS-BaseNotification) v1.3," Oct. 2006
34. Organization for the Advancement of Structured Information Standards (OASIS), "Web Services Resource Framework (WSRF) v1.2," April 2006
35. World Wide Web Consortium (W3C), "Web Service Description Language Version 2," March 2006
36. World Wide Web Consortium (W3C), "Simple Object Access Protocol (SOAP) Version 1.2," April 2007
37. World Wide Web Consortium (W3C), "Web Services Policy Framework (WS-Policy) Version 1.5," Sept. 2007, available online: <http://www.w3.org/TR/ws-policy/>
38. World Wide Web Consortium (W3C), "Web Services Policy Attachment (WS-PolicyAttachment) Version 1.5," Sept. 2007, available online: <http://www.w3.org/TR/ws-policy-attach/>