Chapter 5

Network service discovery, selection, and brokerage

Guanhong Tao[1] and Zibin Zheng[1]

The next generation Internet is expected to cope with new challenges to support a wide spectrum of network applications with highly diverse requirements due to the coexisting heterogeneous network environment. One of the challenges to achieve this objective lies in enabling network domain collaboration and network application interaction without exposing the internal structure and implementation details of each domain, where network virtualization will play a pivotal role in allowing a large number of service providers to offer various network services upon shared network infrastructure. Service-Oriented Architecture (SOA) [39] offers an effective architectural principle for heterogeneous system integration and provides a promising approach to support network virtualization, which can be applied in network service discovery, selection, and brokerage for the special requirements of future Internet. Due to the heterogeneity of network systems in ubiquitous and pervasive computing environments, network service discovery, selection, and brokerage face one of the main challenges to specify network demands of various applications. A key to solve this problem lies in flexible and effective interactions among the heterogeneous networks, various implementations and ubiquitous architectures with scalable information update, network-platform-independent methods, multi-attribute decision-making techniques, etc.

5.1 Introduction

With the rapid development of network architectures and technologies, the Internet is facing great challenges to support the wide spectrum of network applications based on such diversity of heterogeneous network systems. As the various network requirements are proposed by numerous network applications, it has motivated the appearance of new network technologies providing all kinds of functionalities. Existing network architectures, however, do not have the capability of satisfying the requirements of all the network applications completely. The next generation Internet

[1]School of Data and Computer Science, Sun Yat-sen University, China

is facing the fact that there will be various heterogeneous network systems coexisting on the Internet, where the collaboration among these network systems becomes a urgent and significant problem to address.

As to facilitating the collaboration among heterogenous network systems, a great deal of research work has been conducted in both academia and industry. Turner *et al.* [36] provided an exposition of the diversified Internet concept to address the problem of heterogeneous networks. Feamster *et al.* [16] presented an architecture, Cabo, to separate infrastructure providers and service providers in order to support the next generation Internet. DRAGON [25] proposed by Lehman *et al.* allowed dynamic provisioning of network resources by using a distributed control plane across heterogeneous network technologies. These approaches aiming to address the problem of heterogeneous network architectures and technologies were all based on the idea of network virtualization. Network virtualization can decouple and abstract network resources into independent functional components, which can subsequently be discovered, selected, and composed to support various network applications. As an approach first proposed to evaluate new network architectures, network virtualization has shown its capability to collaborate heterogeneous network systems for the next generation Internet.

Service discovery, selection, and brokerage in network virtualization environments face the challenge to effectively and efficiently interact with different network applications. The Service-Oriented Architecture (SOA) [39] is a system architecture designed to coordinate computational resources among heterogeneous systems. These resources are virtualized into services that can subsequently be discovered, selected, and composed to support various computing applications. Such characteristic of SOA can be utilized in network virtualization to address the problem of heterogeneous network systems. Based on the principles of SOA, heterogeneous network systems can be virtualized into network services. The network services are the reusable network functional components decoupled from various heterogeneous network systems. Therefore, the requirements of network applications can be met by discovering, selecting, and composing the existing network services.

Performance-based discovery and selection are the main topics widely studied. Commonly, the requirements of network applications include network performance parameters, such as network bandwidth, delay time, and load balance. The network service broker needs to find the optimal network service that meets the requirement of network applications. Therefore, the capabilities of network services should be evaluated and predicted as to compare them with the requirements of network applications.

In this chapter, the network virtualization concept and its importance for the next generation Internet are discussed. Based on network virtualization, the principle of SOA is introduced to cope with the challenge of various heterogeneous network systems. By adopting the idea of SOA, network resources can be decoupled into independent reusable network components, and then virtualized into network services, which can subsequently meet various network requirements from numerous network applications through service discovery, selection, and brokerage. Network-as-a-Service (NaaS) can considerably facilitate the development of the next generation Internet.

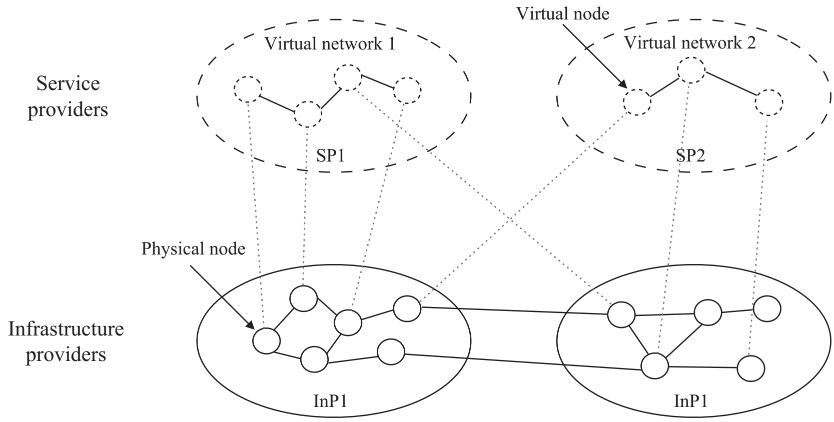

Figure 5.1 Network virtualization environment

5.2 Network virtualization for next generation Internet

With the explosion of various heterogeneous network systems, the next generation Internet is facing the fact that these network systems will coexist, and support various network requirements from numerous network applications. The diversity of network systems, however, is challenging the capability of the current Internet architecture as to support various network applications. Therefore, in order to collaborate diverse network systems, it is urgent and significant to develop new network architectures to flexibly interact with various heterogeneous networks. Network virtualization decouples the roles of the traditional Internet Service Providers (ISPs) into two entities: Infrastructure Providers (InPs) who manage the physical infrastructure, and Service Providers (SPs) who create virtual networks to offer end-to-end services by aggregating resources from various InPs [6]. It can effectively and efficiently address the problem of cooperating with heterogeneous network systems.

Infrastructure providers manage and maintain the physical network infrastructure, and provide physical resources [11]. Differing from the traditional Internet service providers, resources provided by infrastructure providers are employed by service providers instead of end users. Service providers, subsequently, lease the physical resources from various infrastructure providers and construct different kinds of virtual networks for end users. Each service provider manages and maintains one virtual network, which utilizes the resources allocated in the underlying infrastructure [13]. Separating infrastructure and service providers effectively decouples the correlation between network functions inside the same infrastructure provider, and virtualize the physical resources into standard services for end users. As the physical resources decoupled and virtualized into services, virtual nodes based on these resources can be connected by virtual links to form a virtual topology, which is finally composed into the virtual network. Each virtual node can be one particular function provided by the network system or one whole network system. The virtual links present the paths connected between different virtual nodes, where network resources are transferred along these virtual links.

As shown in Figure 5.1, network virtualization separates the traditional Internet service providers into infrastructure providers and service providers. The infrastructure providers provide physical network infrastructures, and InP1 and InP2 are both infrastructure providers. The physical network infrastructures provided by InP1 and InP2 are decoupled into several independent components, which can be utilized by service providers to form virtual networks. SP1 and SP2 are service providers who manage and maintain virtual network 1 and virtual network 2, respectively. The dotted line between virtual nodes in virtual network and physical nodes in InPs denotes the physical resources that service providers lease from infrastructure providers. The solid line between virtual nodes are the resource transfer paths between different virtual nodes. These virtual nodes connected with each other form network services, which then can be employed by end users, i.e., network applications.

A great deal of research work has been conducted on network virtualization in both academia and industry. An exposition of the diversified Internet concept was proposed by Turner *et al.* [36] to address the problem of heterogeneous networks. Similarly, the architecture, Cabo *et al.* [16], separated infrastructure providers and service providers to support various network requirements based on shared physical infrastructures. To facilitate the coexisting approaches and architectures, 4WARD [30] developed a systematic and general approach to network virtualization, which was also investigated by Cabo. Dynamic composition of network resources is also a significant problem that researchers aim to address. DRAGON [25] proposed by Lehman *et al.* allowed dynamic provisioning of network resources by using a distributed control plane across heterogeneous network technologies. Belqasmi *et al.* [2] introduced a novel networking paradigm, Ambient Networks, to enable on-demand and transparent cooperation between heterogenous networks. In the article, they illustrated that three degrees of composition were possible: network interworking, control sharing, and network integration, where network interworking was the most common degree in real situations. Douville *et al.* [8] proposed a service plane architecture that can automatically composite multi-domain network services. More research work about network virtualization is summarized by Chowdhury *et al.* [6].

Diverse and purpose-built proprietary appliances and various network devices make it increasingly difficult to add new services into current networks and raise the network ossification problem. Network function virtualization (NFV) was recently proposed to address these problems and to reduce capital expenditures (CapEx) and operating expenditures (OpEx). According to ETSI white paper [5], NFV decouples the functionality from hardware equipment in specific locations to provide software that can run on industry standard servers, switches, and storage anywhere. By consolidating equipment and workload, NFV can reduce equipment costs, power consumption, as well as energy consumption. Due to the virtualization of hardware in different locations, targeted and tailored services can be rapidly introduced and scaled based on customer needs and geography. Although Han *et al.* [21] pointed out that NFV may bring several challenges to network operators, the application of NFV can greatly benefit the development of NaaS for next generation Internet.

Network virtualization hides the diversify of heterogeneous network infrastructures from end users by providing on-demand network services to meet various requirements of network applications. Differing from the traditional network environments, the next generation Internet is facing the fact that a great number of heterogeneous network systems coexist and the collaboration between these network systems cannot be avoided. Hence, virtualizing the physical resources and abstracting them into services become urgent and inevitable, where network virtualization plays a crucial role in.

Due to the migration from traditional Internet environments to the next generation Internet, one of the most significant technologies, i.e., service discovery and selection techniques, cannot be applicable in network virtualization scenarios. The major reason is that traditional Internet is based on the Internet Service Providers (ISPs) who provide physical resources directly. In network virtualization, however, ISPs is separated into two entities: Infrastructure Providers (InPs) and Service Providers (SPs). The physical resources are provided by InPs, and they are virtualized and abstracted into services managed and provided by SPs. As end users interact directly with SPs instead of ISPs, the discovery and selection techniques that cope with physical resources can be inapplicable in dealing with virtual networks. Besides, traditional service discovery and selection techniques are based on the access performance generated by local access network infrastructures [13]. As network services are virtualized from the physical infrastructures, the network access is no longer limited to the local infrastructures. That is, the performance criteria adopted for discovery and selection should be adapted and improved as to meet the wide perspective of network services.

The great challenge of network service discovery and selection in network virtualization is to interact with numerous heterogeneous network systems and to meet various requirements of network applications. In order to address this problem, researchers have made a lot of efforts and the SOA is widely employed by them. The SOA is an effective architecture for heterogenous system integration, which provides a promising mechanism to facilitate service discovery and selection in network virtualization. The SOA principles and its application in network virtualization is discussed in the next section.

5.3 Service-oriented architecture in network virtualization

SOA is a system architecture initially employed to coordinate computational resources among diverse heterogenous systems to meet various computing requirements of applications in distributed computing environment. As described in article [3], an SOA might be an application architecture within which all functions are defined as independent services with well-defined invokable interfaces. In the SOA, heterogeneous resources are virtualized into services, which can subsequently satisfy the requirements of all kinds of applications through service discovery and selection. The virtualized services are self-constrained and modular, and contain entities that can be used to cooperate with other services. The loose-coupling mechanism is the pivotal feature of SOA that allows services and applications to flexibly interact with each other. Employing this feature of SOA in network virtualization can effectively and efficiently address the problem of the coexistence of numerous heterogeneous network systems.

SOA is widely employed in the Web service research [14,29,41]. As illustrated in the article [22], SOA in Web service includes two features: interfaces with Internet protocols and messages in machine-readable documents. According to Figure 5.2, there are three parties in the SOA system: service provider, service broker, and service customer. The service provider provides service for customers. As there are a plenty of service providers and various requirements from service customers, a service provider needs to publish its service description at the service registry. Service descriptions are organized in standard format that contain functions and access methods. In Web service, the service description is standardized by Web Service Description Language (WSDL) [40]. The service registry is maintained by the service broker that provides various services including service discovery, service selection, service composition, etc. If a service customer (e.g., a Web application) requests a service, it sends its criteria to the service broker. After the service broker receive the query, it applies service discovery and selection algorithms (e.g., Universal Description Discovery and Integration (UDDI) [31]) to search for the optimal service that matches the criteria of the service customer in the service registry. If there exists a matching service, the service broker returns the location and interface information to the service customer. With the service information, the service customer can invoke the service based on the service description. In contrast to the traditional Web service techniques, Simple Object Access Protocol (SOAP) [7], Representative State Transfer (REST) defined by Fielding [17] is much more lightweight. The REST architectural style includes four principles: resource identification through Universal Resource Identifier (URI) [28], uniform interface, self-descriptive messages, and stateful interactions through hyperlinks [32]. These principles guarantee the simpleness of RESTful Web services as REST utilizes existing well-known W3C/IETF standards (HTTP, XML, URI, MIME), which makes the construction of services low-cost and easy-to-build. Due to its stateless feature that provides possibility of serving a very large number of clients, RESTful techniques is promising for realizing the SOA principle and the development of NaaS.

The pivotal feature of SOA, the loose-coupling mechanism, makes SOA applicable in the environment of numerous coexisting heterogenous network systems. A great deal of research work has been carried out in applying SOA. PlanetLab [23] is a global research network that supports the development of new network services, which firstly adopted the virtualization concepts into network services. A slice is a network of virtual machines that run on the computer notes of PlanetLab. Each slice is isolated from each other and allowed to use by different members independently. GENI [15] addressed the shortcomings of PlanetLab and allowed the virtualized network to maintain its own management protocols. Similar to GENI, FEDERICA [34] leveraged virtualization with appropriate hardware and software capabilities of the infrastructure. FEDERICA provided virtual networks via a centralized decision

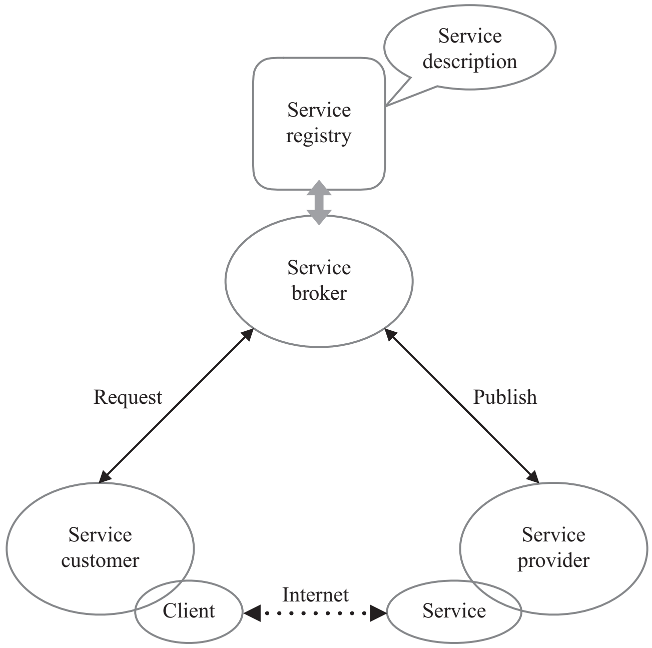

Figure 5.2 The service-oriented architecture in Web service

making procedure, which allowed users to fully configure and manage the resources without affecting the physical infrastructure. UCLPv2 [19] was designed as an SOA that provided virtual networks for users to build their own services or applications. Based on UCLPv2, there was no need for users to deal with the complexities of the underlying network infrastructure. Additionally, articulated private networks (APNs) was presented upon UCLPv2, which can be considered as a next generation virtual private network (VPN). SOA has been considered state-of-the-art techniques for service-delivery platforms [26], which is widely employed in the telecommunications industry. The research work conducted on SOA in telecommunications is summarized by Griffin *et al.* [20].

Applying SOA in network virtualization is to utilize the advantage of the loose-coupling feature of SOA. SOA virtualizes the resources into services, and the paradigm of SOA is illustrated in Figure 5.2. Similar to Web services, network resources can also be encapsulated into network services, and managed and provided by network service providers. The descriptions of various network systems are published in the service registry, which can be maintained by network service brokers. The request from network service customers (e.g., network applications) can be sent to the network service brokers, and then brokers can discover and select the optimal network services that meet the requirements of customers. A layered structure of service-oriented network virtualization proposed by Duan *et al.* [13] is shown in Figure 5.3. This structure separates the traditional structure of Internet

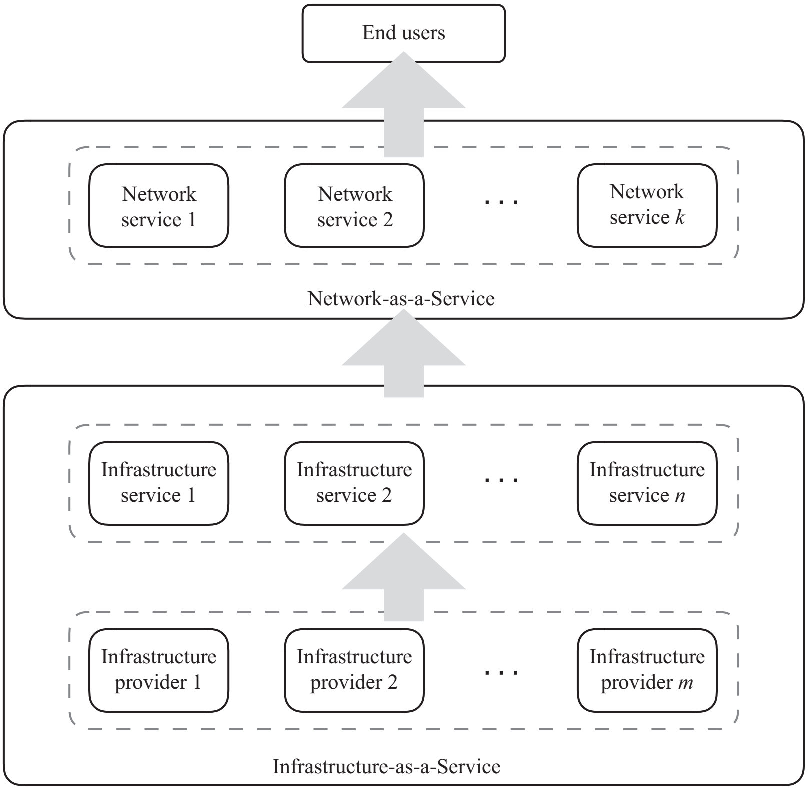

Figure 5.3 The structure of service-oriented network virtualization

service providers into two layers: a Network-as-a-Service layer and an Infrastructure-as-a-Service layer. The two layers are composed by service providers discussed in the previous section of network virtualization. Infrastructure providers virtualize the physical network resources into infrastructure services, where these services are the base of Infrastructure-as-a-Service. Based on the infrastructure services, network service providers construct various network services according to the requirements of end users (e.g., network applications). Therefore, end users can utilize the underlying network resources by accessing the network services provided by service providers. This constructs the Network-as-a-Service paradigm.

By adopting the principle of SOA in network virtualization, the physical network resources are decoupled and virtualized into independent network services. It addresses the problem of the diversity of numerous network systems, which may lead to the complexity of the interaction between service providers and end users. Based on SOA paradigm, the requirements of end users can be satisfied by discovering, selecting, and composing the network services provided by service providers. Hence, in order to assist network users to find the optimal network services that meet their requirements, the techniques of network service discovery, selection, and brokerage need to be proposed.

5.4 Network service discovery and selection

The research problem of network service discovery and selection has attracted a great attention from academia and industry [4,24,38]. As to assisting the network service broker to discover and select optimal network services that meet the requirements of network applications, performance-based discovery and selection techniques have be carried out by researchers. Based on the requirements of network applications, the discovery and selection procedure can be conducted via performance prediction, which is the major concern of network applications. After predicted the performance of network services, sufficient amount of network resources needs to be allocated as to guarantee network provisioning.

5.4.1 Performance prediction for network service discovery and selection

For the network service broker, the network service discovery and selection is based on the performance requirements of network applications, the characteristic of network traffic generated by network applications, and the provisioning capabilities of available network services [12]. The performance requirements and the characteristic of network traffic are provided by network applications during the request process. Therefore, the capabilities of network services should be evaluated and predicted by the network broker as to seek for the optimal network services that meet the requirements of network applications.

Due to the rapid development of Internet, the next generation Internet is facing the burst of various network applications. As there exists a vast number of network applications that demand various network services, proposing a common approach that describes the requirements is of urgency and significance. In order to cope with this important problem, Duan *et al.* [12] defined a *Demand Profile* $\mathbf{P}(\mathbf{d}, \mathbf{L}, \mathbf{a})$ that presented the general specification of network requirements, where $\mathbf{d}$ denoted the address set, $\mathbf{L}$ was a traffic load descriptor, and $\mathbf{a}$ indicated the performance requirement set. Let $T^{in}(t)$ be the accumulated amount of network traffic generated by a network application by the time t. The network application that has a nondecreasing, nonnegative function $A(\cdot)$ for any nonnegative time t and s satisfies the following:

$$T^{in}(t) - T^{in}(s) \leq A(t - s). \tag{5.1}$$

This equation gives the upper bound of the amount of network traffic that a network application may load on the network service. Commonly, the arrival traffic generated by network applications is shaped by the traffic regulation mechanism [27,37] using leaky buckets [1,35]. Hence, the arrival curve is defined as follows:

$$A(t) = \min\{pt, \sigma + \rho t\}, \tag{5.2}$$

where p is the peak rate, σ is the maximal burst size of the traffic flow, and ρ is the sustained rate.

As to discovering and selecting network services for network applications that meet their requirements, the performance of network services needs to be predicted based on the *Demand Profile* $\mathbf{P}(\mathbf{d}, \mathbf{L}, \mathbf{a})$. The performance prediction methods were proposed by previous research work [9,10]. Let $S(t)$ denote the service curve of the network route R provided by a network service. Then the minimum bandwidth guaranteed by the network service on the network route R can be defined as:

$$b_{min} = \lim_{t\to\infty} \left\lfloor \frac{S(t)}{t} \right\rfloor. \tag{5.3}$$

Assume arrival curve $A(\cdot)$ denotes the traffic load descriptor $\mathbf{L}$ in *Demand Profile*, then the maximum delay d_{max} for data transportation can be calculated by the following equation:

$$d_{max} = \max_{t\geq 0}\{\min\{\delta : \delta \geq 0 \;\&\; A(t) \leq S(t+\delta)\}\}. \tag{5.4}$$

The article [9] also proposed the performance prediction on Latency-Rate (LR) server [33]. Assume the arrival curve of the traffic load generated by a network application is $A(t)$ defined in (5.2), and the service curve of the network route R is $S(t) = r(t-\theta)$, where r and θ are the service rate and latency of the network flow, respectively. Therefore, the minimum bandwidth guaranteed by the network service on the network route is:

$$b_{min} = \lim_{t\to\infty} \frac{r(t-\theta)}{t} = \lim_{t\to\infty} \left(r - \frac{r\theta}{t} \right) = r. \tag{5.5}$$

Based on (5.4), the maximum delay for data transportation guaranteed by the network route can be calculated as:

$$d_{max} = \theta + \left(\frac{p-r}{p-\rho}\right)\frac{\sigma}{r}, \quad r \geq \rho. \tag{5.6}$$

With the predicted performance of a network service on a network route R defined in (5.3) and (5.4), network brokers can discover and select network services that meet the requirements of network applications. As the requirements of network applications are presented in the performance requirement set $\mathbf{a}$ of the *Demand Profile* $\mathbf{P}$, the discovery and selection procedure is based on the comparison between the predicted performance and the required one. The article [12] conducted the following analysis and proposed a service selection approach. The request of network services generated by a network application can be classified into three categories: (1) With only bandwidth requirement, i.e., $\mathbf{a} = b_{req}$; (2) with only delay requirement, i.e., $\mathbf{a} = d_{req}$; (3) with both bandwidth and delay requirements, i.e., $\mathbf{a} = \{b_{req}, d_{req}\}$. Therefore, based on the request of network applications, the selection process is decided as follows: (1) If $\mathbf{a} = b_{req}$, select network services with $b_{min} \geq b_{req}$; (2) if $\mathbf{a} = d_{req}$, select network services with $d_{max} \leq b_{req}$; (3) if $\mathbf{a} = \{b_{req}, d_{req}\}$, select network services with $b_{min} \geq b_{req}$ and $d_{max} \leq b_{req}$. Network services that meet the performance requirements of network applications can be selected as the candidate services. If there are more than one network services selected as candidate services, then other criteria such as service cost or load balance will be adopted to select the optimal network service [10].

5.4.2 Resource allocation for network service provisioning

As to making sure that allocated network resources satisfy the network provisioning for the network application, the selected network service ought to allocate adequate amount of resources, e.g., the bandwidth of underlying network systems. In order to address this problem, research work in articles [10,12] discussed the bandwidth allocation for Quality of Service (QoS) provisioning in network services. Detailed discussion is presented in the following paragraphs.

Analyzed in the article [10], the delay performance is in the scope of $[\theta, \theta + \frac{\sigma}{\rho}]$, and the service rate is in the range of $[\rho, p]$. Therefore, the optimal allocation of bandwidth is to find the most reasonable service rate r_a that satisfies the delay requirement d_{req} given by the network application, which is defined as:

$$r_a = \min\{r : \rho \leq r \leq p \ \& \ d_{max}(r) \leq d_{req}\}. \tag{5.7}$$

In order to meet the requirement of network applications, the delay performance guaranteed by the network service should not be greater than the required delay performance d_{req}, which is to satisfy the following function:

$$d_{max} = \theta + \left(\frac{p-r}{p-\rho}\right)\frac{\sigma}{r} \leq d_{req}. \tag{5.8}$$

Therefore, the minimum bandwidth that ought to be allocated to satisfy the requirement of the network application can be obtained as:

$$r_a = \frac{p\sigma}{(p-\rho)(d_{req}-\theta)+\sigma}. \tag{5.9}$$

From (5.9), it can be seen that the service rate r_a is upper bounded by the peak rate p, and it approaches the peak rate when the sustained rate ρ gets close to the peak rate. The sustained rate denotes the current condition of the network, and if the sustained rate is near the peak rate, it means that the network is under the best condition with the full utilization of resources.

To summarize the allocation of bandwidth for guaranteeing a delay requirement d_{req}, here gives a viable function:

$$r_a = \begin{cases} \rho & d_{req} \geq D_{max} \\ \frac{p\sigma}{(p-\rho)(d_{req}-\theta)+\sigma} & D_{min} \leq d_{req} < D_{max} \\ \text{no valid value} & d_{req} < D_{min} \end{cases}, \tag{5.10}$$

where $D_{max} = \theta + \frac{\sigma}{\rho}$ and $D_{min} = \theta$. If the requirement of the network application includes both bandwidth and delay performances, i.e., $\mathbf{a} = \{b_{req}, d_{req}\}$, the minimum bandwidth allocation can be obtained by the following function:

$$b_{min} = \max\{r_a, b_{req}\}. \tag{5.11}$$

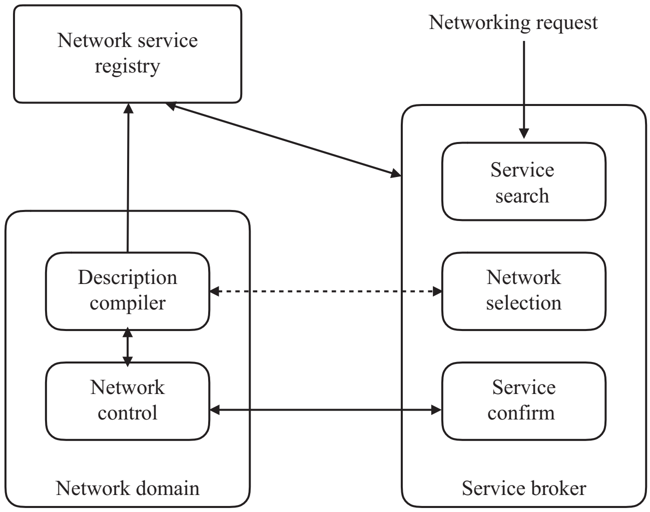

Figure 5.4 The structure of network service broker system

5.5 Network service brokerage

The SOA changes the relation between network end users and network service providers by introducing the network service broker. The purpose of network service broker is to register network services based on the service description provided by network service providers, and to discover and to select appropriate network services according to the requirements of network end users. With the network service broker, network end users can submit their various requirements to the broker and do not have to consider the underlying infrastructures provided by network service providers. Similarly, the network service providers virtualize the network resources into different types of network services in the same format, and publish them on the service registry with unified service descriptions. Therefore, network end users and network service providers do not have to be compatible and seek for each other as the network service brokers mediate between them. To be more specific, the article [9] gave a detailed analysis of networks service brokerage, and a network service broker system is shown in Figure 5.4.

The service broker provides the functionalities of network service discovery and selection. To illustrate the whole structure of network service broker system, network domain and network service registry are also illustrated in Figure 5.4, which are significant components in providing and supporting network services. Network domain is the part where the network service provider interacts with the network to publish service descriptions and build network connections. As the network fluctuates all the time, the network information provided by network service providers is not stable and invariant. Therefore, the description compiler is employed to collect the current network status and generate the network service description. Based on the current network status, the network service description can be published on the network service registry. The network control component copies with the connection between network service providers and end users, which is at the last stage of the whole process. The service broker is the main and significant part in the interaction between network providers and network end users. When a networking request sent from a network end user, the network service discovery and selection process is conducted by the service broker. Firstly, based on the requirement submitted by the end user, the service broker searches the registered network services in the network service registry to find appropriate network services. As analyzed in Section 5.4.1, the request **P(d, L, a)** includes the destination address **d**, the traffic load descriptor **L**, and the performance requirement **a**. Those network services that satisfy the *Demand Profile* **P** are selected as candidates for further selection. The dotted line between description compiler and network selection denotes the selection process based on the current network status. That is, by checking the current network status, the service broker selects the optimal network service under the current network condition from the network service candidates. After selecting the satisfied network service, the service broker check with the network control to confirm resource allocation for network service provisioning. Once the connection between the network service provider and the end user is established, then the network service can be utilized by the network end user.

5.6 Information update for network service discovery and selection

The next generation Internet can be a large scale dynamic network environment with various network services provided different network service providers. With the highly dynamic update of service information, it is urgent and significant to keep the latest service information available at the network service registry, which is the foundation of network service discovery and selection. But the update of service description may occupy a large part of network resources between the network service provider and the network service broker/registry if the entire service description is republished at the registry. In order to address this problem, an efficient and scalable protocol for network service information update was proposed in the article [11].

The protocol proposed in the article [11] was based on subscription-notification mechanism, where one-way notification messages are sent by providers to multiple information consumers. A standard approach for notification was provided by the Origination for the Advancement of Structured Information Standards (OASIS) [18], which employed a topic-based publish/subscribe mechanism. By adopting this approach, the network broker/registry can subscribe a certain set of network states as topics to network services. Then the update of network service information can be completed by only changing the subscribed network states, which can greatly enhance the update process of network service information and reduce the frequent communication between the network service provider and the network service broker/registry. Figure 5.5 proposed by Duan *et al.* [11] presents the detailed updating process.

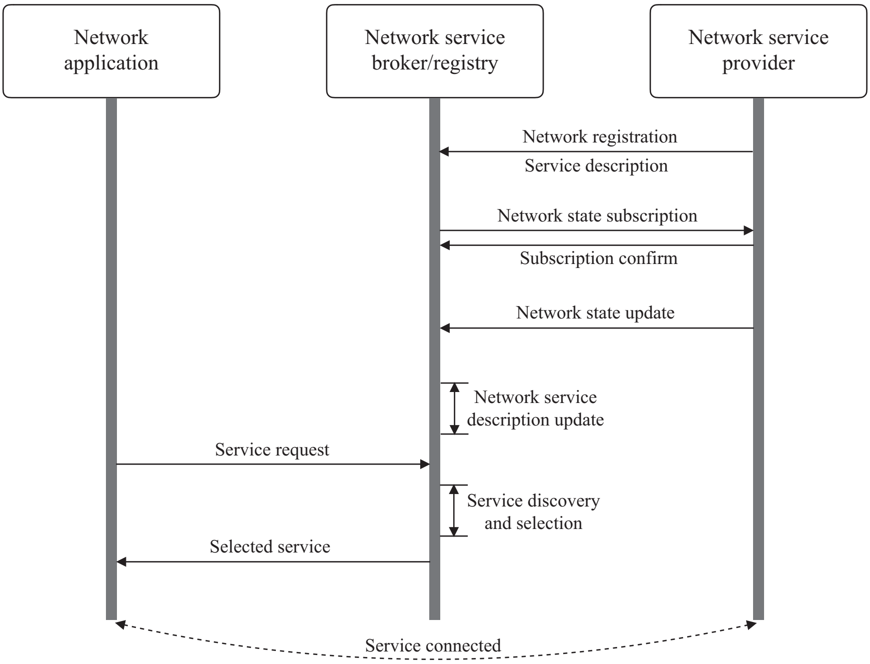

Figure 5.5 Network service information update

The network service provider publishes a service at the network service registry with a service description document. According to the service description, the network service registry can subscript interested network states to network service providers by sending a subscription notification. The network service provider confirms the subscription of the network service registry and updates the scripted network states. After the network service broker/registry subscribes the specified network states of network services, the notification can be pushed to the network service broker/registry whenever there is change occurred on network services. The network service broker/registry updates the network service description based on the notification message including network state update. Therefore, if a network application sends a service request to the network service broker/registry, the network service discovery and selection process can be carried out according to the latest network information. The most appropriate network service that meets the requirement of the network application is selected and sent to the network application (ID of the selected service) by the broker. The dotted line represents the connection between the network application and the network service provider, and the network application starts accessing the selected network service.

Dynamic update of network information is significant for the discovery and selection of network services due to the instability of Internet and network services.

In order to provide the optimal network service that meets the requirement of network applications, the broker has the responsibility to choose the most appropriate one from thousands of network services, which shows the best performance at that specific time. The update procedure implements the description compiler discussed in Section 5.5.

5.7 Conclusions

The next generation Internet has to face the fact of the coexistence of various heterogenous network systems. Due to the rapid increase of network requirements from all types of network applications, it is crucial to provide network functionalities without considering the underlying infrastructures. In order to address this critical problem, network virtualization provides the capability to decouple and abstract network resources into independent functional components. The underlying infrastructures are provided and managed by the infrastructure providers, and the service providers provide network services for network end users by utilizing the virtualized underlying network resources. The technique of network virtualization is widely adopted by researchers in solving the problem of coexisting heterogenous network systems and show its capability to collaborate these network systems. As to cope with the diversity of requirements of network applications, SOA is introduced and applied in the network virtualization environment. The key feature of SOA is the loose-coupling mechanism, which makes SOA applicable in the environment of numerous coexisting heterogenous network systems. The network services provided by service providers can be published and managed by the service brokers, which makes it convenient and efficient for network applications to acquire the network services that meet their requirements. The communication between network applications and network service providers is surrogated by network service brokers who discover and select the most appropriate network service for network applications. Therefore, the network service discovery and selection can be conducted in the service brokers. The request from network applications include the requirements of network performance. Due to the instability of Internet, the performance of network services needs to be predicted based on the service description under the corresponding circumstance. Additionally, sufficient network resources should be allocated to satisfy the network provisioning for the network applications. Network service brokerage optimizes the communication between network applications and network service providers. The architecture and technologies proposed and discussed by previous researchers are network-independent, which are applicable to heterogenous network environments of the next generation Internet.

References

[1] Leaky bucket. https://en.wikipedia.org/wiki/Leaky_bucket.

[2] Fatna Belqasmi, Roch Glitho, and Rachida Dssouli. Ambient network composition. *IEEE Network*, 22(4):6–12, 2008.

[3] Kishore Channabasavaiah, Kerrie Holley, and Edward Tuggle. Migrating to a service-oriented architecture. *IBM DeveloperWorks*, 16, 2003.

[4] Liang Chen, Yipeng Feng, Jian Wu, and Zibin Zheng. An enhanced QoS prediction approach for service selection. In *Proceedings of the Eighth IEEE International Conference on Services Computing (SCC'11)*, pages 727–728, 2011.

[5] Margaret Chiosi, Don Clarke, Peter Willis, *et al.* Network functions virtualisation: An introduction, benefits, enablers, challenges and call for action. In *SDN and OpenFlow World Congress*, pages 22–24, 2012.

[6] NM Chowdhury and Raouf Boutaba. Network virtualization: State of the art and research challenges. *IEEE Communications Magazine*, 47(7):20–26, 2009.

[7] Francisco Curbera, Matthew Duftler, Rania Khalaf, William Nagy, Nirmal Mukhi, and Sanjiva Weerawarana. Unraveling the Web services web: An introduction to SOAP, WSDL, and UDDI. *IEEE Internet Computing*, 6(2):86, 2002.

[8] Richard Douville, J Le Roux, J Rougier, and Stefano Secci. A service plane over the PCE architecture for automatic multidomain connection-oriented services. *IEEE Communications Magazine*, 46(6):94–102, 2008.

[9] Qiang Duan. Service-oriented network abstraction and discovery for the diversified Internet. In *Proceedings of the IEEE Sarnoff Symposium*, pages 1–5, 2007.

[10] Qiang Duan. Network service description and discovery for high-performance ubiquitous and pervasive grids. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 6(1):3, 2011.

[11] Qiang Duan. Automatic network service discovery and selection in virtualization-based future Internet. *Network Protocols and Algorithms*, 4(2):30–48, 2012.

[12] Qiang Duan and Enyue Lu. Network service description and discovery for the next generation Internet. *International Journal of Computer Networks (IJCN)*, 1(1):46–65, 2009.

[13] Qiang Duan and Yong Lu. Service-oriented network discovery and selection in virtualization-based mobile Internet. *Journal of Computer Information Systems*, 53(3):38–46, 2013.

[14] Thomas Erl. *Service-oriented architecture: A field guide to integrating XML and web services*. Upper Saddle River, NJ, USA: Prentice-Hall PTR, 2004.

[15] D. Clark, S. Shenker, A. Falk. GENI Research Plan. Technical Report, GENI Research Coordination Working Group, April 2007.

[16] Nick Feamster, Lixin Gao, and Jennifer Rexford. How to lease the Internet in your spare time. *ACM SIGCOMM Computer Communication Review*, 37(1):61–64, 2007.

[17] Roy Thomas Fielding. *Architectural styles and the design of network-based software architectures*. PhD thesis, University of California, Irvine, 2000.

[18] Steve Graham, David Hull, and Bryan Murray. Web services base notification 1.3. *Organization for the Advancement of Structured Information Standards (OASIS)*, 2006.

[19] Eduard Grasa, Gabriel Junyent, Sergi Figuerola, Albert Lopez, and Michel Savoie. UCLPv2: A network virtualization framework built on web services [web services in telecommunications, part II]. *IEEE Communications Magazine*, 46(3):126–134, 2008.

[20] Dayton Griffin and Dirk Pesch. A survey on web services in telecommunications. *IEEE Communications Magazine*, 45(7):28–35, 2007.

[21] Bo Han, Vijay Gopalakrishnan, Lusheng Ji, and Seungjoon Lee. Network function virtualization: Challenges and opportunities for innovations. *IEEE Communications Magazine*, 53(2):90–97, 2015.

[22] Hao He. What is service-oriented architecture. *Publicação eletrônica em*, 30:50–55, 2003.

[23] Scott Karlin and L Paterson. PlanetLab: A blueprint for introducing disruptive technology into the Internet. In *joint Princeton ACM/IEEE Computer Society meeting*, 2003. https://www.planet-lab.org/.

[24] George Lee, Peyman Faratin, Steven Bauer, and John Wroclawski. A user-guided cognitive agent for network service selection in pervasive computing environments. In *Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications (PerCom'04)*, pages 219–228, 2004.

[25] Thomas Lehman, Jerry Sobieski, and Bijan Jabbari. DRAGON: A framework for service provisioning in heterogeneous grid networks. *IEEE Communications Magazine*, 44(3):84–90, 2006.

[26] Thomas Magedanz, Niklas Blum, and Simon Dutkowski. Evolution of SOA concepts in telecommunications. *Computer*, 40(11):46–50, 2007.

[27] Domingo Marrero, Elsa M Macías, and Alvaro Suárez. An admission control and traffic regulation mechanism for infrastructure WiFi networks. *IAENG International Journal of Computer Science*, 35(1):154–160, 2008.

[28] Larry Masinter, Tim Berners-Lee, and Roy T Fielding. Uniform resource identifier (URI): Generic syntax. 2005.

[29] Eric Newcomer and Greg Lomow. *Understanding SOA with web services (independent technology guides)*. Boston, MA, USA: Addison-Wesley Professional, 2004.

[30] Norbert Niebert, Stephan Baucke, Ibtissam El-Khayat, *et al.* The way 4WARD to the creation of a future Internet. In *Proceedings of the IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'08)*, pages 1–5, 2008.

[31] Organization for the Advancement of Structured Information Standards (OASIS). Universal Description, Discovery and Integration (UDDI) v3.0.2. https://www.oasis-open.org/standards.

[32] Cesare Pautasso, Olaf Zimmermann, and Frank Leymann. Restful web services vs. 'big' web services: Making the right architectural decision. In *Proceedings of the 17th international conference on World Wide Web*, pages 805–814, 2008.

[33] Dimitrios Stiliadis and Anujan Varma. Latency-rate servers: A general model for analysis of traffic scheduling algorithms. *IEEE/ACM Transactions on Networking (ToN)*, 6(5):611–624, 1998.

[34] Peter Szegedi, Sergi Figuerola, Mauro Campanella, Vasilis Maglaris, and Cristina Cervelló-Pastor. With evolution for revolution: Managing FEDERICA for future Internet research. *IEEE Communications Magazine*, 47(7):34–39, 2009.

[35] Jonathan Turner. New directions in communications (or which way to the information age?). *IEEE communications Magazine*, 24(10):8–15, 1986.

[36] Jonathan S Turner and David E Taylor. Diversifying the Internet. In *Proceedings of the Global Telecommunications Conference (GLOBECOM'05)*, volume 2, pages 1–6, 2005.

[37] Jing Wang and Suparerk Premvuti. Distributed traffic regulation and control for multiple autonomous mobile robots operating in discrete space. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'95)*, volume 2, pages 1619–1624, 1995.

[38] Shangguang Wang, Zibin Zheng, Qibo Sun, Hua Zou, and Fangchun Yang. Cloud model for service selection. In *Proceedings of the 30th IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 666–671, 2011.

[39] Sanjiva Weerawarana, Francisco Curbera, Frank Leymann, Tony Storey, and Donald F Ferguson. *Web services platform architecture: SOAP, WSDL, WS-policy, WS-addressing, WS-BPEL, WS-reliable messaging and more*. Upper Saddle River, NJ, USA: Prentice-Hall PTR, 2005.

[40] World Wide Web Consortium (W3C). Web Services Description Language (WSDL) Version 2.0. https://www.w3.org/TR/wsdl20/.

[41] Zibin Zheng and Michael R Lyu. Collaborative reliability prediction of service-oriented systems. In *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering (ICSE'10)*, pages 35–44, 2010.