Toward Self-Managed Media Stream Processing Service Overlays

Xiaohui Gu, Philip S. Yu
IBM T. J. Watson Research Center
Hawthorne, NY 10532
{xiaohui, psyu}@ us.ibm.com

Abstract

On-demand media stream processing service provisioning on top of a service overlay network (SON) has emerged as a promising approach to providing quality-aware and failure-resilient media streaming services. Although previous work has addressed different problems in media streaming, it is still an open problem to manage SON under dynamic stream environments. In this paper, we propose a novel model-based adaptive SON management framework that can achieve better QoS-aware media stream processing services than unmanaged SONs. We first describe a statistical SON model that can capture the popularity of different service functions and the interaction patterns between different services. We then present the model-based SON management framework and use the SON topology manager as a case study to illustrate the main idea. We are implementing a prototype of the self-managed SON framework and our initial experimental evaluation shows promising results of our approach.

1 Introduction

Today's Internet has become an important service delivery infrastructure rather than merely provides host connectivity. Various Internet services such as video streaming, voice-over-IP, Internet TV have been widely deployed and used. Traditionally, Internet services are provisioned in a centralized monolithic way. Each application service is hosted as a whole on a single computer or computer cluster. Users can access the application service via Internet. However, this centralized monolithic approach is hardly scalable and cannot be customized based on user's requirements. To address the above problems, dynamic service composition [3, 8, 1, 9, 5, 6, 4] has emerged as a promising approach to achieving scalable and efficient service provisioning. To support efficient composite service delivery with quality-of-service (QoS) provisioning, service overlay networks (SONs) [5, 6] have been proposed to provide the underlying infrastructure for QoS-aware service composition. The SON connects distributed hosts into an overlay network on top of existing Internet infrastructure. Each host can provide one or more media stream processing components such as video correlation [7] and transcoding. The QoS-aware service composition can be conveniently performed on top of the SON, which can dynamically map abstract service requests (i.e., service templates) into instantiated composite services (i.e., service graphs) based on the user's function and QoS requirements. With SON, individual component service provider can publish their service components by joining the SON and allow other users to utilize their service components. The QoS-aware service composition can intelligently select among replicated service instances based on the QoS and resource information maintained by the SON. Moreover, SON provides a resilient routing infrastructure for quality-aware service composition by maintaining multiple redundant overlay paths between two SON nodes.

Although previous work has addressed different aspects in QoS-aware service composition, it is still an open problem to provide autonomic management infrastructure for the SON itself. Since the QoS-aware service composition is performed on top of the SON, the quality of the SON has a direct impact on the quality of upper-level composed services. In this paper, we propose a novel self-managed SON framework that can perform model-based continuous optimizations on the SON adapting to both the upper-level service provisioning activities and underlying IP network changes. Our main idea is to dynamically maintain a statistical *SON model* and perform *informed* SON adaptations based on the knowledge provided by the SON model. Specifically, the SON model can capture the popularity of different service functions and the interaction patterns between different service components. Each SON management module can make proper adaptation decisions based on the dynamically maintained SON model. The goal of the SON management tier is to automatically optimize the SON so that we can efficiently deliver QoS-aware media stream processing services on top of the SON. We are implementing a prototype of the self-managed SON framework and

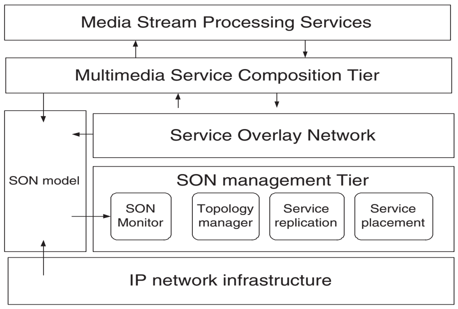

Figure 1. Self-managed SON for QoS-aware media stream processing services.

our initial experimental evaluation shows promising results of our approach.

The rest of the paper is organized as follows. Section 2 introduces overall system architecture of the self-managed SON framework. Section 3 presents the details of the SON model and the model-driven SON topology management. Section 4 presents some initial experimental results. Finally, the paper concludes in Section 5.

2 System Architecture

In order to achieve efficient QoS-aware media stream processing services, we need to provide various dynamic system information (e.g., QoS state of different service instances, load conditions of different hosts) to the service composition module and adaptively adjust the SON structure to better satisfy user's service requirements. To achieve this goal, we propose to add a SON management tier into the system, illustrated by Figure 1. The SON management tier consists of four major modules:

- *SON monitoring module* is responsible for collecting various dynamic QoS and resource information about service instances, overlay hosts, and overly links. The SON monitoring module can resolve information queries from the service composition tier or other SON management modules for efficient service delivery.

- *SON topology manager* controls the mesh topology of the overlay network. The SON topology manager can dynamically update the neighbor set of each overlay node according to workload and environment changes. For scalability, each overlay node is usually connected to a limited number of neighbors into a bounded-degree overlay mesh. Since the QoS-aware service composition is performed on top of the SON, the quality of the overlay mesh has a direct impact on the quality of service delivery. Thus, the SON topology should be carefully managed to achieve not only failure resilience but also good QoS provisioning.

- *service replication manager* decides how many replicated service instances should be allocated to each provisioned service function. The replicated service instances can share workload given many concurrent service requests and survive system node and link failures. The number of service instances represents the trade-off between QoS provisioning and replication overhead. To maximize the overall system utility under the resource constraints, we need to carefully decide the number of service instances for each supported service function.

- *service placement* manager decides where those replicated service instances should be placed. In the SON, the composite service delivery consists of multiple service instances that are located at different distributed hosts. Thus, the relative "distances" between these service instances can greatly affect the QoS provisioning of the composed service delivery.

The goal of the SON management tier is to automatically optimize the SON so that the service composition tier can more efficiently provide QoS-aware media stream processing services based on user's service requirements. We believe that the SON management tier must be adaptive based on two key observations. First, the service request workload can be highly dynamic. Different users can request different service functions at various time. For example, a certain service function can be in higher demand than other services at a particular time. Moreover, the support of service composition makes the service request workload even more volatile since users can request different combinations within the same set of service functions. Second, the SON infrastructure is Internet-based distributed service infrastructure, which makes it inherently dynamic. For example, the service hosting overlay nodes and underlying IP network connections can experience transient overload conditions or provisioning failures. Thus, any static SON scheme can be insufficient to achieve efficient quality-aware service delivery under dynamic service requests and hosting environments.

Our main idea is to make SON management tier adaptive, which can continuously optimize the SON based on environment changes to achieve most efficient QoS-aware media stream processing service. To achieve effective SON adaptations, we propose a model-driven approach to SON management, where a dynamically maintained SON model is used as the knowledge-base for our adaptive decisions. Based on well-maintained SON model, we can make *informed* adaptations to avoid introducing inefficiency and instability into the system.

3 Model-driven SON Management

In this section, we present the model-driven SON management framework. We first introduce the major SON model elements which encapsulate the statistical inter-service interaction patterns and service popularity ranks. We then describe the adaptive SON topology management as a case study to illustrate the idea of model-driven SON management.

3.1 SON Model Elements

The SON model is dynamically derived based on the media stream processing service activities in the current SON environment. The SON model consists of two major statistical system properties:

- *Service popularity ranks* characterize the demand for different service functions by the current service request workload. The service popularity can be used to direct the adaptations of SON monitoring and service replication modules. For example, we can perform more frequent state update on the instances of popular service functions and less or no state update on the instances of unpopular service functions. The service replication module can create more instances for heavily requested service functions and reduce instances for not so popular service functions.

- *Inter-service interaction patterns* describe the inter-service relationship such as the frequency of interactions and traffic volume between two service instances. The SON topology management and service placement modules can make intelligent adaptation decisions based on the interaction pattern information. For example, the topology manager can make two frequently communicating overlay hosts direct neighbors. The service placement module can migrate two heavily interacting service instances onto the same host for reducing network communication cost.

The SON model elements are represented in a statistical fashion based on a history of recent service requests and service composition results. We use two time-series to represent dynamically arrived service requests and instantiated service graphs. We denote a sliding-window on each time-series, which consists of recent service requests or service graphs within a certain time period (e.g., last five hours). The service popularity and interaction pattern models are continuously updated according to the time-series data. For example, when a new service request $\xi = F_1 \rightsquigarrow F_2$ arrives, it is inserted into the sliding-window and replaces an old service request $\xi' = F_3 \rightsquigarrow F_4$. The SON model manager then increases the occurrence of service functions F_1 and F_2 by one and decreases the occurrence of F_3 and F_4 by one. The popularity of each service function is calculated by the ratio between its occurrence number over total occurrence number of all service functions. Similarly, when the service composition module instantiates a new service graph in the SON, the service graph is inserted into the sliding-window of its time-series and retires an old service graph. The SON model manager then updates the interaction frequency and traffic volume between different service instances according to the new service graph and the retired old service graph.

3.2 SON Topology Management

For a large-scale SON, fully connected mesh is not a practical solution since each node needs to maintain a large number of neighbors. Thus, large-scale SON typically adopts a bounded-degree overlay mesh where the system puts an upper-bound on the number of neighbors each node can has. The overlay mesh with limited node degree may cause QoS degradation due to the overlay stretch effect [2]. If two overlay nodes v_i and v_j are not directly connected in the overlay mesh, the data transmission between v_i and v_j needs to traverse an overlay path consisting of multiple overlay links. Thus, the overlay network QoS (e.g., delay, loss rate, bandwidth) may become worse than the direct IP network connection. Since the media stream processing services are provided on top of the overlay network, the overlay stretch has a direct impact on the QoS of the composed service. The goal of our overlay topology adaptation algorithm is to minimize the overall overlay stretch for all current service sessions.

We define that a host v_j is "service-bounded" with the host v_i if the connection from v_i to v_j is used by at least one service session. We define the bounding degree of the host v_j to v_i as the number of the service sessions that include the connection from v_i to v_j in their service graphs[1]. Suppose each overlay node v_i can have at most d out-bound and in-bound neighbors. For all the hosts that are service-bounded to v_i, we can calculate their bounding degree based on the interaction patterns captured in the SON model. We always select the hosts that have highest bounding degrees with v_i to form the neighbor set of v_i. For example, let us consider a host v_j that is not included in the neighbor set of v_i. If v_j has a higher bounding degree than at least one existing neighbor of v_i, and v_j can accept an extra inbound neighbor, v_j is added into the out-bound neighbors of v_i if the out-bound neighbor set of v_i is not full. Otherwise, v_j replaces an existing neighbor v_k of v_i that has the lowest bounding degree with v_i. The host v_k then deletes v_i from its inbound neighbor set.

[1] We can also use other weighting functions such as the network traffic volume.

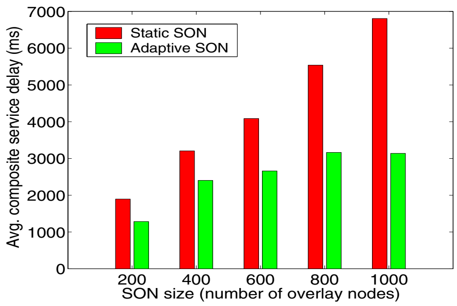

Figure 2. Effect of adaptive SON management.

4 Experiment Results

We are implementing a prototype of adaptive SON management tier in the SpiderNet service composition system [6]. The SpiderNet service composition system can perform dynamic service composition under multiple QoS constraints on top of an arbitrary SON. We have finished an initial implementation of the adaptive SON monitoring module and SON topology management module. All the other management modules are still under development. As a proof of concept, we have conducted some initial experiments to test the effectiveness of the implemented management modules. We compare the QoS-aware service composition performance on top of an adaptive SON tier and a static SON tier, illustrated by Figure 2. We gradually increase the SON size from 200 nodes to 1000 nodes with a fixed node degree $d = 10$. Thus, larger system sizes can have bigger overlay stretch due to an increasing number of hops between every two nodes since the probability that two overlay nodes are connected is reduced. The static SON uses a fixed mesh topology where each host randomly selects 10 other hosts as neighbors. The neighbor set is not updated during runtime. In contrast, the adaptive SON dynamically updates the SON topology to adapt to service composition workload changes. We observe that the adaptive SON can consistently improve the quality of composed services in terms of reduced end-to-end service delays. The above experimental results show that adaptive SON management can greatly improve the performance of QoS-aware service compositions.

5 Conclusion

In this paper, we have presented a novel model-based adaptive SON management framework to achieve efficient QoS-aware service composition. The goal of the SON management tier is to adaptively adjust the SON structure to achieve better QoS-aware composite service delivery under current workload conditions. To achieve this goal, the system first maintains a statistical SON model that can capture the popularity of different service functions and interaction patterns between different service instances. The SON management modules can thus make informed adaptation decisions based on the SON model. Specifically, we present an adaptive SON topology management scheme that can adjust the neighbor set of each overlay node based on the inter-service interaction patterns. Our work represents the first step toward a fully adaptive SON management infrastructure for providing QoS-aware multimedia stream processing services. We are implementing the prototype of the SON management infrastructure and our initial experimental evaluation shows promising results of our approach.

References

[1] A. P. Black, J. Huang, J. Walpole, and C. Pu. Infopipes: an Abstraction for Multimedia Streaming. *Multimedia Systems (special issue on Multimedia Middleware), 8(5), pp.406-419*, 2002.

[2] Y. Chu, S. G. Rao, S. Seshan, and H. Zhang. A Case For End System Multicast. *IEEE Journal on Selected Areas in Communication (JSAC), Special Issue on Networking Support for Multicast*, 2002.

[3] X. Fu, W. Shi, A. Akkerman, and V. Karamcheti. CANS: Composable, Adaptive Network Services Infrastructure . *Proc. of 3rd USENIX Symposium on Internet Technologies and Systems*, March 2001.

[4] X. Gu and K. Nahrstedt. On Composing Stream Applications in Peer-to-Peer Environments. *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, 2006.

[5] X. Gu, K. Nahrstedt, R. N. Chang, and C. Ward. QoS-Assured Service Composition in Managed Service Overlay Networks. *Proc. of IEEE 23nd International Conference on Distributed Computing Systems (ICDCS 2003)*, 2003.

[6] X. Gu, K. Nahrstedt, and B. Yu. SpiderNet: An Integrated Peer-to-Peer Service Composition Framework. *Proc. of IEEE International Symposium on High-Performance Distributed Computing (HPDC-13), Honolulu, Hawaii*, June 2004.

[7] X. Gu, Z. Wen, C.-Y. Lin, and P. S. Yu. ViCo: An Adaptive Distributed Video Correlation System. *In Proc. of ACM Multimedia (SIGMM)*, 2006.

[8] P. K. Mckinley, U. I. Padmanabhan, N. Ancha, and S. M. Sadjadi. Composable Proxy Services to Support Collaboration on the Mobile Internet. *IEEE Transactions on Computers, 52(6)*, June 2003.

[9] B. Raman and R. H. Katz. Load Balancing and Stability Issues in Algorithms for Service Composition. *Proc. of IEEE INFOCOM 2003, San Francisco, CA*, April 2003.