Semantic-Enhanced Distribution & Adaptation Networks

Bo Shen, Zhichen Xu[†], Susie Wee, and John Apostolopoulos
Multimedia Communications and Networking Department, Hewlett-Packard Labs.
[†]*Yahoo! Inc.*

Abstract

Recent years have witnessed significant efforts in deriving and embedding semantic information in content for improved content retrieval, adaptation, and distribution. Relatively little work considers leveraging this semantic information in the infrastructure to better serve the needs of clients and achieve better cost-effectiveness of infrastructure resources. In this paper, we identify semantics that can be derived and extracted from the various components in a content distribution infrastructure, namely content semantics (from content source), infrastructure semantics and client semantics (from content consumer). We develop a semantic-enhanced distribution and adaptation framework (SEDAN) that achieves superior efficiency and provides content access and adaptation features that were not previously possible.

1. Introduction

In recent years there has been a significant research thrust in identifying important semantic information for media content and creating media analysis tools that automatically derive this information. Content-based semantic information has a large scope ranging from high-level semantic objects that are meaningful to humans (e.g., identifying people in a scene) to low-level semantic objects that are more directly related to the sampled media signals (e.g., color, shape, or texture). Different applications require different types of semantic information about the media, leading to the large scope of this data. Various attempts have been made to organize this content-based semantic information, including the MPEG-7 standard as well as various proprietary systems.

Another trend is that users are consuming media content through an increasingly large diversity of clients connected over heterogeneous networks. Matching the media content to the capabilities of each requesting client and its network connection requires semantic information about the client and network capabilities. For the client this may include (1) hardware attributes such as processing, storage, and display size, (2) software attributes such as OS, available media decoders and DRM solutions, (3) network connection attributes including network type, access bandwidth and QoS, and (4) access privileges for pay-per-access or confidentiality. A variety of mechanisms for identifying and communicating these client semantics to the sender, as well as client/server negotiation, have been developed in the ITU, MPEG-21, as well as in proprietary systems.

A content distribution and adaptation network, hereafter referred to as the infrastructure, can both deliver and adapt media content for the requesting client. To accomplish this goal in a manner that scales to support the large number of clients while providing high performance for each client, it is highly desirable to build intelligence into the infrastructure that provides various types of semantic information. We call this a semantic-enhanced distribution & adaptation network (SEDAN). SEDAN stores information such as available services and resources, their properties, and their locations relative to the requesting clients, as well as measured network characteristics that can be used to build a network weather map. SEDAN uses this information to drive content delivery and adaptation as a function of time.

SEDAN delivers content from content creators to content consumers as shown in Figure 1. It differs from traditional CDNs in two ways. First, it uses mid-network services to adapt content in the network to facilitate end-to-end content delivery. Second, it intelligently manages content delivery and adaptation services by maintaining semantic information about the content, the infrastructure and the clients.

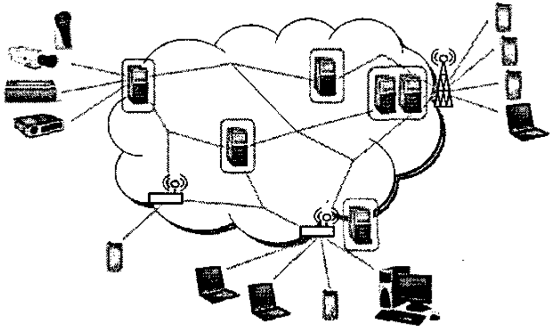

Figure 1: SEDAN delivers original and adapted content from content creators to diverse clients over time-varying network links. A service overlay provides compute and storage resources that facilitate content delivery, storage, and mid-network adaptation. A management fabric or knowledge plane stores semantic information about the content, client, and infrastructure; this intelligence can be used by the various system components.

At SEDAN's core is a management fabric or knowledge plane that provides intelligence to the system and allows communication between components through semantic information sharing. For example, a network layer service component that measures the network proximity between clients and overlay nodes can semantically insert proximity information into the knowledge plane; this information can be accessed by a management service component that assigns content adaptation sessions to "the best" overlay nodes for a given client. The knowledge plane also tracks service characteristics in the infrastructure, including service node location, service description, and resource availability. Furthermore, the knowledge plane can track content as it moves and adapts across the infrastructure and it can track statistics of user requests and client device capabilities as they evolve over time. This information can be used by content providers so they can decide what types of content

0-7803-8603-5/04/$20.00 ©2004 IEEE

they should publish in the infrastructure. For example, the content owner may adjust offerings with knowledge that 90% of its clients are on handheld devices and only 10% on desktop computers.

2. Framework

We envision SEDAN to provide the following two key components: (1) a knowledge plane, and (2) a decision plane. Other components update and query the knowledge plane about the existence of content, services, and resources. The decision plane provides intelligent decision making such as selecting which content server to serve content or which transcoding server to adapt content.

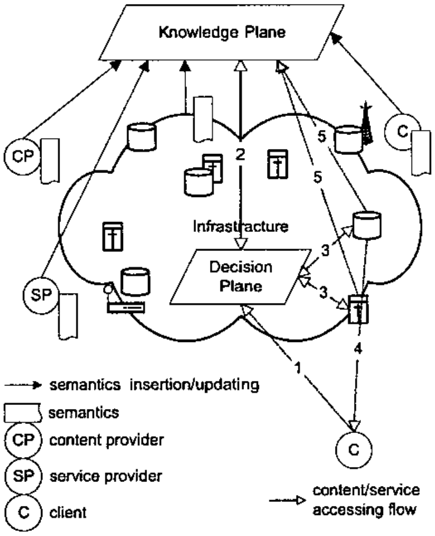

Figure 2: Components, participants, and processing flow in SEDAN.

2.1 Knowledge Plane

The knowledge plane has several fundamental problems to address, including (1) how to precisely capture, store, and organize the semantics of the various kinds of information, (2) how to handle the potentially millions of entities that are in the system, and (3) how to handle various dynamics, such as changing user preferences, service and resource availability, and network conditions.

To tackle the data organization problem, SEDAN represents the various types of semantic information using a formally defined data model. Two important features of the data model include generality and the ability to handle dynamic schemata evolution.

The data model should be generic enough to handle different kinds of semantic information. For content, common types of semantic information may include (1) file versioning; (2) application-based dependencies, e.g., movie *A* is based on novel *B*; (3) attributes such as the author and creation time of content, (4) content-based semantics, e.g., feature vectors of text documents, edge distribution, or color distribution of images; (5) context-based information, e.g., scenes that belong to the same movie.

For services, the semantic information includes the following. (1) The functional aspect of the service, e.g., a description of the service, its input/output specifications, and constraints on the inputs/outputs. For example, based on functional description, SEDAN can determine that both a transcoder and a keyframe extractor can be used to reduce the bit rate. (2) The environmental requirements, which may include the hardware and its capacity, the operating system, and software libraries and packages; (3) the inter-dependence among services, from which we can determine how to compose multiple services to satisfy a given client request. For resources, the semantic information may include network conditions, locations of network resources, hardware configurations, and software installations. After a service is deployed onto a resource, it also includes more dynamic information such as the current load and available capability. For clients, the knowledge plane manages user profiles (e.g., preferences and buddy lists) as well as device capabilities.

SEDAN addresses scalability considerations by grouping related entities according to locality in network proximity, semantics, and client interests. For example, SEDAN groups information about resources and services close to each other in terms of network-layer proximity metrics (e.g. delay) by using a landmark measurement technique such as Netvigator [2]. To find the nearest node to a given node in a scalable manner, Netvigator applies a clustering algorithm using the local and global landmark information.

Other considerations include consistency and availability of the knowledge plane as well as security and access control to it. To make the information available, SEDAN replicates much of the semantic metadata in the knowledge plane. To ensure the consistency of the data, SEDAN enforces it based on a publish/subscribe model, where a replica is a subscription to the master copy, and clients subscribe to each of the replicas. This way, each client and replica can have an individualized consistency policy. To ensure the authenticity and integrity of the metadata, SEDAN authenticates each component in the system and protects the data with standard data protection techniques.

Very little prior work has addressed these considerations for implementing such a knowledge plane. Distributed query processing such as PIER [5] was proposed to improve the DHT query facilities based on a traditional SQL-like API. PIER has focused on monitoring distributed networked resources, but has paid little attention to capturing and organizing semantic information of various kinds of entities. SEDAN's knowledge plane can be implemented as described in Pstore [1]. Pstore provides a framework of basic relations and tools for generating and using semantic metadata. At the core of Pstore is a generic data model that is based on the Resource Description Framework (RDF) [6].

Given the data model, the knowledge plane provides support for (1) basic relations for various entities, (2) change management via a publish/subscribe model, (3) customized views to group related entities together to present them to different clients, (4) security and access control to ensure the authenticity of data and data protection, (5) advanced searching capabilities so other components can search for services/contents/resources based on multiple criterion. With the help of the semantic knowledge stored in the knowledge plane, intelligent decision-making can be a powerful tool to facilitate semantic-based content distribution.

2.2 Decision Plane

Realizing effective delivery and adaptation requires solving a number of fundamental design problems including: (1) where to deploy the servers that provide storage, processing, and delivery capabilities (server placement), (2) how to distribute the media content and media processing services across the servers (content/processing distribution), and (3) how to select for each client the best servers hosting the content for media delivery and associated services (server selection).

SEDAN performs distributed decision making by consulting the information stored in the knowledge plane. The functionality of the decision plane includes (1) enabling system administrators to input policies (in the form of distributed algorithms); (2) enabling client requests by providing semantic-enhanced APIs. To tackle the scalability problem of being able to handle large numbers of users, SEDAN distributes the decision process by identifying the relevant entities according to network proximity and having the decision nodes closest to the entities (clients, service nodes, content providers) make the decisions.

3. Operations

The logical definition of the knowledge and decision planes results in a much richer set of interactions within SEDAN than the traditional CDNs.

3.1 Knowledge insertion and updating operations

The insertion of the knowledge (semantics) is performed in two steps: setup and utilization. SEDAN's knowledge plane provides semantic insertion APIs for four participating entities, namely for the content provider, service provider, infrastructure provider, and clients. For example, when a content provider begins hosting a piece of content, it can inform the knowledge plane of the content semantics. Or when a transcoding server is added into the infrastructure, the knowledge plane is informed of its location and connectivity characteristics. Alternatively, a client can also register its connection and/or device profile with the knowledge plane.

Semantic information that is dynamic, such as client usage patterns or the real-time load of a transcoding server, should also be updated in the knowledge plane. Once again, SEDAN provides APIs for each participating entity. End user agents or edge servers can be in charge of the updating process by collecting information such as access rate and location of the flash-crowd. Content providers (or its agents in SEDAN) can update the knowledge plane with the popularity of the content. The infrastructure can report resource utilization such as load on service boxes and traffic within the distribution network. In addition, there can be agents deployed inside the infrastructure to extract semantics for content on its delivery path and insert that in the knowledge plane.

3.2. Decision making process

The semantic information gathered in the knowledge plane as described above is very useful in helping the decision plane make intelligent decisions on how the content or services are delivered. Figure 2 also shows a content access and adaptation flow. (1) Upon receiving of a client request, (2) the decision plane consults the knowledge plane for the client profile and the semantics of the requested content. (3) The decision plane then sets up the source server and the adaptation service and (4) delivers the requested content to the client. (5) The usage of both the content and the service is then updated in the knowledge plane.

Intelligent decision making is enabled by the aggregated semantic stores in the knowledge plane. Specifically, the SEDAN framework enables: (1) Content-based retrieval based on semantics aggregated from the entire content distribution environment. (2) Semantically-accurate content adaptation under resource constraints, i.e., an approximate adaptation that provides semantically accurate adaptation. (3) Semantically-optimized service dispatching.

4. SEDAN Examples

Multi-point Conferencing We give an example of SEDAN that improves the utilization and capacity of the network resources for multi-user video/audio conference sessions. In this scenario, users are located in different parts of the network. When a conference call is requested, a multi-point control unit (MCU) is assigned to the session. Media streams from all the users are sent to the MCU, which then combines the streams and sends an aggregate stream to all the users.

Users have buddy lists of people with whom they typically communicate. During various parts of the day, a person may have different communities with whom he is likely to communicate. For example, on weekends and evenings the person may be likely to communicate with friends and family members; while during work hours the person may be more likely to communicate with his work colleagues. Even during work hours, the most likely communication members may change, e.g., a US worker is more likely to have calls with Europe in the morning and with Asia in the evening.

While typically each person may have a fixed MCU through which they route their chaired calls, SEDAN allows the MCU to be selected dynamically based on the semantic information. This dynamic selection is performed in a manner that optimizes the use of network and system resources. At a first level, SEDAN uses network information to select the best MCU server based on the participant locations. Server selection is optimized for the session by calculating all the participant locations using a network proximity tool such as the one discussed in Section 5. In addition, client location information for the session can be extracted in advance through other content modalities such as email invitations that include lists of participants by inserting this information into the knowledge plane as discussed in Section 3.1.

In some scenarios, the conference participants may not be known in advance. In this case, SEDAN can enhance MCU selection by using the chairperson's profile information to predict which other users are likely to join and where they are likely to be located.

In another scenario, SEDAN uses all the user profiles to predict when and where heavy loads may occur at different times, and selects the best MCU server based on this aggregate knowledge stored in the knowledge plane. For example, SEDAN may choose a second-best MCU server for a session based on a prediction that many other sessions are likely to occur between the US and Asia in the evening.

In the following illustrative example, we simulate multipoint conference scenarios in which the MCUs and conference participants are chosen from nodes in PlanetLab. We use actual measured network latency among the nodes. This information is stored in the knowledge plane. In addition, a buddy list for each user is simulated and also stored in the knowledge plane as semantic information. We compare the average link latency for a conference initiated by each user. Three MCU selection methods are used. The first one uses random selection without the help of semantic information. The second method chooses the MCU nearest to the conference initiator. The third method further uses the buddy list semantics to choose an optimal MCU for the conference. Figure 3 shows that the semantic-enhanced MCU selection always generates the best conference experience.

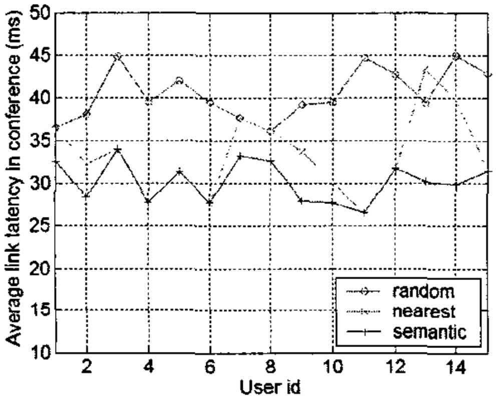

Figure 3: Semantic-enhanced MCU selection.

Semantic-accurate Content Adaptation A simple rule-based policy enables semantically-accurate content adaptation under resource constraints. In an example scenario, the client profile indicates that the client needs low-delay and bit rate adapted delivery. The decision plane identifies that the requested origin content residing in the content provider site is too distant to satisfy the low-delay requirement of the request. However, a nearby replica server stores semantically similar content, so the replica is chosen as the content source. In addition, the decision plane also determines that the full-fledged bit rate transcoding box is fully loaded at the moment. Fortunately, the infrastructure semantics indicate that a key-frame extraction service box is available and also lies along the delivery path. Therefore, the key-frame extraction service is chosen instead of the bit rate reduction service.

Optimizing Aggregate Application-level Quality Given a set of resources in the network, as well as measurements of the network conditions, the infrastructure should automatically and dynamically decide how to operate in order to maximize the system performance. Performance is measured using different metrics that focus on end-user or infrastructure quality, e.g. application-level quality as seen by the clients, or total number of clients that the infrastructure supports. Measuring and optimizing application-level quality requires accurate metrics for predicting the application-level quality as a function of the infrastructure operation. For example, when streaming video over a best effort network the received quality is a function of the bandwidth, delay, and losses. Accurate models for predicting video quality as a function of network characteristics have been proposed, and these models enable the infrastructure to accurately predict the quality that a client receives for a given path/node characteristics, and enable the infrastructure to select the best server to stream or service the content for each client. Furthermore, these tools allow one to optimize more complicated service offerings, by accounting for the effects the resource selection and network characteristics have on the aggregate application-level quality as seen by the clients.

5. Summary

We have presented a framework of a semantic-enhanced distribution and adaptation network system. We have also examined the benefits of using semantic information in various media distribution and adaptation scenarios. We have developed a system that uses knowledge and decision planes for networked media delivery and media services [3][4]. This architecture uses a service location manager that monitors network conditions and server load, then recommends the best server for a given media service request. The next step is to distribute the knowledge and decision planes and effectively capture and organize the various kinds of semantic information.

6. References

[1] Z. Xu, M. Karlsson, C. Tang and C. Karamanolis, "Toward a semantic-aware file store," The 9th Workshop on Hot Topics in Operating Systems, May 2003.

[2] Z. Xu, P. Sharma, S-J Lee and S. Banerjee, "NetVigator, scalable network proximity estimation," Hewlett-Packard Labs. Technical Report, HPL-2004-28, Feb. 2004.

[3] M. Harville, M. Covell, S. Wee, "An architecture for componentized, network-based media services," Proc. ICME'03, July 2003.

[4] S. Wee, J. Apostolopoulos, W. Tan, S. Roy, "Research and Design of a Mobile Streaming Media Content Delivery Network," IEEE ICME, July 2003.

[5] R. Huebsch, J. M. Hellerstein, N. Lanham, B. T. Loo, S. Shenker, I. Stoica, "Querying the internet with pier," 29th *Very Large Data Base Conference* (VLDB), 2003.

[6] W3C. "Resource description framework (rdf) model and syntax specification", www.w3.org/TR/REC-rdf-syntax, Feb. 99.