RC 21709 (97781) 28 March 2000

ComputerScience/Mathematics

00A000197

IBM Research Report

Distributed Application Service for Internet Information Portal

Chung-Sheng Li, John R. Smith, Rakesh Mohan, Yuan-Chi Chang
IBM T. J. Watson Research Center
P. O. Box 218
Yorktown Heights, NY 10598

Brad Topol, John Hind, Yongcheng Li
IBM Network Computing Software Division
4205 South Miami Blvd, RTP, NC 27709

IBM
Research Division
Almaden - Austin - Beijing - Haifa - T. J. Watson - Tokyo - Zurich

Distributed Application Service for Internet Information Portal

Chung-Sheng Li, John R. Smith, Rakesh Mohan, and Yuan-Chi Chang
IBM T. J. Watson Research Center, P. O. Box, 704, Yorktown Heights, NY 10598
Brad Topol, John Hind, and Yongcheng Li
IBM Network Computing Software Division, 4205 south Miami Blvd, RTP, NC 27709

Abstract

As Internet information portal become prevalent for both Internet and Intranet, most existing Internet Application Server architectures are not scalable to support large amount of personalization, customization and content adaptation required. In this paper, we propose a framework to capture the information and content dissemination process. Furthermore, we propose a methodology to map this process to a distributed application server environment. By fully exploiting the intersections of user preference at multiple content processing stages, this new framework enables high hit ratio on processing, storage, and transmission of content and thus scales well to support a large number of clients.

1. Introduction

Internet information portal has become increasingly prevalent in the past few years due to the rapid advances in networking technologies and substantial increase of the population that have access to Internet. Many Internet portal sites try to serve as the gateway to the Internet. They provide either general information (such as Yahoo, Lycos, Netscape Netcenter, Altavista) or specialized information (such as WebMD.com for health information, ZDNET.com for computer and internet information, auto.com for automotive information, consumerworld.org for consumer related information) to the Internet users. These portals have already attracted a disproportionate amount of Internet traffic and advertising dollars, and have the capability to influence the success of new electronic commerce adventures [6]. Portal service for Intranet has also become increasingly important as an essential mechanism for productivity improvement. The end objective of both Internet and Intranet information portal is to provide a *one-stop overview* of information that is *relevant* to the end user.

Most of these portal sites allow substantial personalization in order to allow customization of the information categories and layout of the site for the end user. Existing Internet Information portal provide the customization at the content server, which may consist of a web server that serves content from a local file system. More sophisticated content server utilizes one or more web servers and one or more application server(s) connecting to a backend database. Currently, application server examples include Netscape application server, Cold Fusion, IBM Websphere, and Oracle application server. This architecture, however, is not scalable to a large number of users distributed over wide range of geographic areas. This is mainly because in the current architectures, processing, storage, and transmission of personalized content require the number of servers to grow *linearly* as the number of clients grows. Furthermore, as Internet appliances such as Palm, PDAs, and smart phones become widely available, massive personalization of Internet Information Portal become a severe challenge to existing Internet infrastructure [1,2].

In this paper, we propose to use data flow diagrams to represent the data processing that is required starting from content providers to the Internet information portal on end devices. The basic operations include data selection, transformation (potentially include both categorization and summarization), merging/ aggregation, and distribution. We also propose distributed application server architecture, and a mapping from the data flow to the distributed application servers. Distributed application server mapping allows the intersections of user preference in multiple content processing stages to be aggregated, thereby making the service architecture scalable.

2. Preliminary

A typical Internet Information portal for a user may involve the selection of multiple information categories from multiple information sources. The information from these sources will be customized based on the user's interests/preferences. Further customization may be required depending on the specific user devices that have been used to access the portal. A typical portal may appear as in Fig. 1.

In this example, the personalized information portal includes breaking news, local news, email, calendar, personalized stock tickers, and the to do list for the project. Each of this interest area may be further divided into multiple sub areas. For example, the breaking news can be categorized into top stories, business, technology, sports, etc. Each of these categories may be provided by a different content provider or data sources with a different update schedule. (Stock ticker may have to be refreshed at a much more frequent rate than calendar.)

For the same individual, personalized information portal may have to be adapted in order to suite different device. For example, text may have to be summarized, image may have to be scaled, converted from color to black-and-white, video may have to be converted to key frames. Speech recognition and synthesis is important for devices such as smart phone. In general, media rich presentations suitable for PC and workstations will have to be adapted through either *modality translation* (video to image, text to speech, and speech to text) or *fidelity transformation* (image scaling, text summarization, etc.) for displaying on pervasive devices [2,3,4,5]. These two types of operations are generally referred to as content adaptation/transformation. Additional device dependent transformation may be required to transform tables, forms, and other type of data in order to view these data on those Internet appliances.

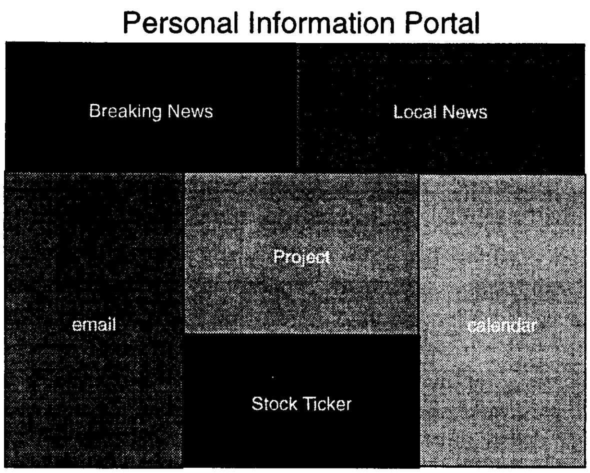

Figure 1 Example of personal information portal.

3. Proposed Method

From the discussions in the previous section, it is apparent that there are four basic operations involved in the workflow for developing an information portal:

- *Select*: For structured data such as tables and databases, this select operation can be an SQL query. For semi-structured data such as XML/HTML documents, this operation can be the selection of a subtree of the DOM (document object model) that corresponds to the XML document. For non-structured data such as collections of images, videos and audios, the select operation could involve content-based retrieval queries utilizing features such as texture, color, and shape information [7,8,9]. Currently, there is an ongoing effort on MPEG-7 to provide standardized metadata description of multimedia data. This will enable multimedia data to be filtered from interoperable repositories. The selection criteria are usually derived from a combination of user interests and end device characteristics.
- *Transformation*: The selected data may need to go through modality, fidelity, or format transformation before it is merged or distributed [2,3,4,5]. Modality transformation may include video to image, text to speech, speech to text, etc. Fidelity transformation may include scaling of image and video, reduction of color, video and text summarization. Format transformation may include the transformation from gif to jpeg, from HTML to WML, etc.
- *Merge/Join*: For structured data, multiple tables may be joined (on one or more attributes) to form a single table. Multiple semi-structured information sources can be merged according to a specific DOM structure and/or layout. Multiple non-structured data sources (images, video, and audio) can also be merged into single media stream.
- *Intermediate cache*: The cache is for storing intermediate results from selection, transformation, and merging. The cache also serves the function as isolating the data flow involving push/publication and the data flow involving pull/subscription.

The entire process of generating information portal for each end user can be represented by data flow diagrams. An example of such a diagram is shown in Fig. 2. In this figure, content from two servers are selected and merged. The merged content, as well as the content selected from three other sources are transformed and merged before being presented to the end user. Note that Fig. 2 only shows the data flow for one end user. For multiple users, there are processing nodes common to subsets of users. These common nodes can exploit multicast and push

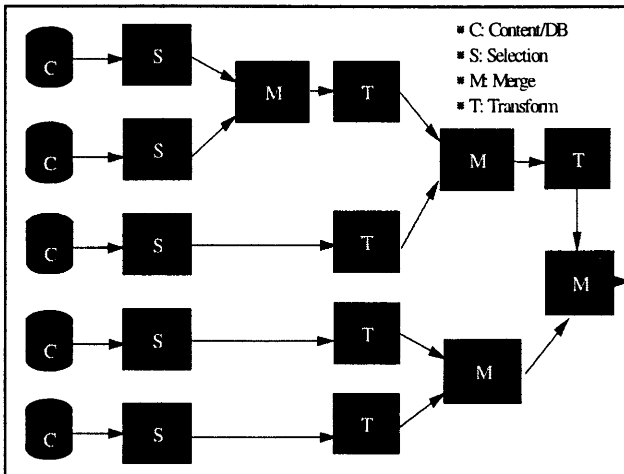

Figure 2 A data flow representation of the information processing workflow from content sources to an end user.

technologies to reduce processing/bandwidth consumption as depicted in Fig. 3. In this case, the content from the content server is decomposed into either non-redundant or partially redundant components. These components are then sent to proxies or intermediate servers. The components from the proxies are then multicast and pushed into the clients. High-end clients may receive multiple components and merge into a single high quality presentation, while low-end device receive only a subset of those components. Additional proxy may be needed in order to perform combining operations for those devices that cannot perform the merging themselves.

An example of such decomposition is progressive videoconference. Video content can be decomposed using progressive video coding (or layered video coding). Each of the video streams is then broadcast/multicast separately. High-end devices will combine multiple stream to form high-resolution full frame rate video playback, while low-end devices will utilize a subset of these streams for low-resolution low frame rate playback. The general process for setting up the personal Internet information portal is as follows:

1. The end user set up the personal preference in terms of information categories. Each of the categories (or subcategories) obtains its information items from one or more information providers.
2. Intermediate Internet proxies will be responsible for requesting content from content providers and merging multiple categories on behalf of the end user.
3. Identical categories from a certain number of users will be grouped together.

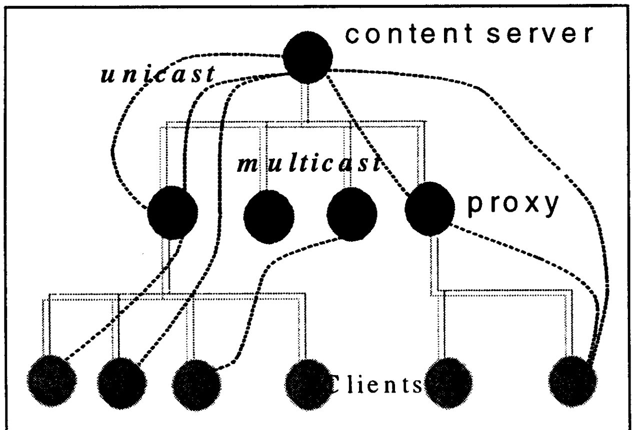

Figure 3 Content dissemination from content server to clients using multicast and push.

Setting up the subscription process of the information portal is similar to the setting up of a network connection such as the set up of an ATM (Asynchronous Transfer Mode) connection. Admission control can be incorporated during this set up process to ensure the resource is not overcommitted so that certain level of quality of service (QOS) can be ensured.

Optimizing Content Distribution Network

This is a two-stage process: The first step is the analysis and optimization of the logical data flow diagram, while the second step is the optimal mapping of the logical data flow nodes to physical application servers that can execute these operations.

The optimization of logical data flow diagram usually involves rearranging data processing steps. The following steps are permissible rearrangements:

- Select/filtering and transformation: These two operations are generally interchangeable. However, there is always a preference of performing filtering in front of the transformation, to avoid the possibility of transporting large amount of data.
- Select/filtering and merging: These two operations cannot be swapped if the Select/filtering operations are scheduled to be performed prior to the merge operations. On the other hand, these two operations can be swapped if the Select/filtering operation is after the merge operation.
- Merging and transformation: If the transformation occurs before the merge, the operation is not exchangeable, as a merge operation combines multiple inputs into a single output. The transformations prior to the merge, unless identical, cannot be swapped with the merge operation. The transformation can be moved before the merge if the transformation is located after the merge.
- The Internet cache can be inserted at any locations.

Once the data flow of the content dissemination process is captured, the next step is to map the data flow into physical application servers, as shown in Fig. 4. The mapping of the data flow operations to application servers involve (1) the computation of the resource requirement for each operation so to (2) determine whether a specific operation need to be mapped to multiple application servers.

Although mapping of content servers from logical to physical is usually one-to-one, the mapping of the selection, transformation, and merging operations to application servers can be one-to-one, one-to-many or many-to-one:

- One-to-one: this is the simplest situation in which each operation is mapped to a single application server.
- One-to-many: in this case, a single operation may be mapped to a cluster of application server either with load balancing or process migration capabilities. This type of mapping is usually considered when the

operation is computational intensive or/and required by lots of end users.

- Many-to-one: in this case, multiple adjacent or non-adjacent operations are mapped to a single application server. This type of mapping is usually considered when the operation is less computational intensive and /or required by few end users.

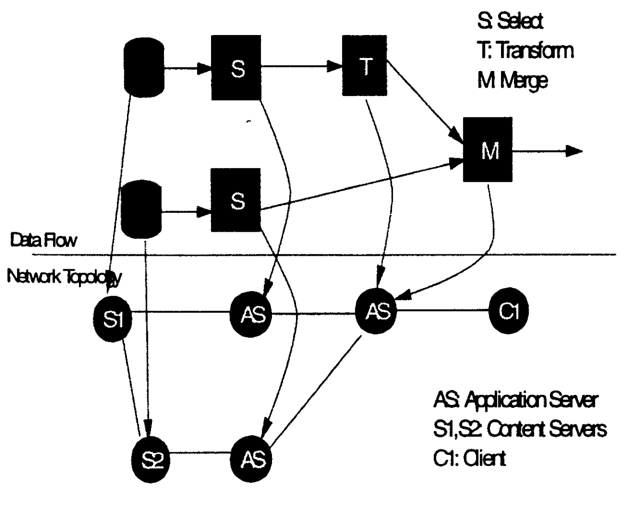

Figure 4. Mapping the logical data flow to the physical network of application servers.

In addition to the consideration of available resources for deploying application services, additional constraints may arise due to the fact that some application service may only be executed on specific application servers. For example, text-to-speech synthesis has to be executed on servers with speech engines.

4. Summary

In this paper, we present a distributed architecture to support an emerging new application – personal information portal for Internet appliances. The basic challenge to this problem is the massive personalization and data transformation required to support large amount of such devices. We propose to use a data flow diagram to capture the data dissemination process from the content servers to the end device. Furthermore, this data flow is mapped to the application servers that are available using a two-stage process: rearrange the data flow in order to minimize the possible communication overhead, and map the data flow to the physical application servers based on the required resource.

5. References

[1] C.-S. Li, R. Mohan and J. R. Smith, "Multimedia Content Description in the InfoPyramid," *IEEE Proc. Int. Conf. Acoustic., Speech, Signal Processing (ICASSP)* , Seattle, WA, Special session on Signal Processing in Modern Multimedia Standards, May, 1998.

[2] J. R. Smith, R. Mohan, C.-S. Li, "Content-based Transcoding of Images in the Internet," *Proceedings of the International Conference on Image Processing (ICIP)*, 1998

[3] J. R. Smith, R. Mohan and C.-S. Li, "Transcoding Internet Content for Heterogenous Client Devices," *Proc. IEEE Inter. Symp. on Circuits, Syst. (ISCAS),* Special session on Next Generation Internet, June, 1998. June, 1998.

[4] J. R. Smith, "Digital Video Libraries and the Internet," *IEEE Communications Mag*, Special issue on the Next Generation Internet, Jan., 1999.

[5] J. R. Smith, "VideoZoom Spatio-temporal video browser," *IEEE Trans. Multimedia*, Vol. 1, No. 2, June, 1999.

[6] R. Dewan, M. Freimer, and A. Seidmann, "Portal Combat: The Battle between Web Pages to become the Point of Entry to the World Wide Web,'' Proc. 32nd Hawaii International Conf. On System Sciences, 1999.

[7] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J.Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker, ``Query by image and video content: The {QBIC} system," IEEE Computer, vol. 28, no. 9, pp. 23 -- 32, September 1995.

[8] J. R. Smith and S.-F. Chang, ``Visualseek: A fully automated content-based image query system," in Proc. International Conference on Image Processing, 1996.

[9] J. R. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, R. C. Jain, and C. Shu, ``Virage: image search engine: an open framework for image management," in Symposium on Electronic Imaging: Science and Technology-Storage & Retrieval for Image and Video Databases {IV}. IS&T/SPIE, 1996, vol. 2670, pp. 76 -- 87.