We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Challenges and the Solutions for Multimedia Metadata Sharing in Networks

Hochul Shin
Samsung Electronics Co., LTD,
South Korea

1. Introduction

Nowadays, multimedia devices such as PCs, smart phones, and televisions have large storage capacities to store a lot of multimedia files and high-speed network ability to connect with each other and the Internet. The demand for multimedia content sharing in networks is scaled up supported by the availability of storage capability and network speed. Thus, innovative infrastructures and technologies have been introduced to efficiently share multimedia contents in distributed systems.

One of the fundamental features of this sharing is metadata acquisition from the multimedia devices, because users in clients need to obtain information about multimedia contents from servers before they decide to manipulate or share the contents. However, obtaining multimedia metadata across multiple servers in networks usually results in high round-trip latency from a request to the response. Since this isn't responsive enough, a user may feel the user interface sluggish, delayed or frozen for significant periods.

Additionally, users expect that the Search function with certain conditions and Sort function by the titles, the artists, the dates, etc. of contents are supported regardless of the location of the contents or the number of servers. However, adding additional protocols to support various kinds of metadata-based service is very difficult to implement and deploy due to the interoperability problem.

Caching responses, historical and statistical information based prefetching, aggressive link based prefetching, full prefetching, and metadata database object acquisition, and the metadata aggregator-based centralized solution have been introduced to resolve these poor responsiveness and limitation of service extensibility from a user's perspective. Some solutions have several usability problems such as long warm-up time or have a lack of service extensibility, while others hurt the interoperability of the multimedia metadata sharing system.

In this chapter we introduce the solutions and guide you in order to figure out their limitations, pros and cons. This chapter is structured as follows: Section 2 covers the multimedia metadata sharing solutions and the examples; Section 3 explains the critical problems of metadata acquisition in multimedia sharing systems; Section 4 introduces the solutions to the problems; Conclusion section briefly restates challenges and the solutions for multimedia metadata sharing in networks.

2. Multimedia metadata sharing solutions

As demand for multimedia content sharing in networks has grown rapidly, innovative infrastructures and technologies have been introduced to efficiently browse multimedia content lists, control playbacks, and deliver multimedia contents in distributed systems.

Fig. 1. A typical multimedia sharing scenario of DLNA

Thus, Universal Plug and Play Audio and Video (UPnP AV) architecture[1] has been introduced as a candidate for multimedia distribution middleware for local networks, becoming widespread in many multimedia devices and software solutions in PCs. This architecture has been supervised by Digital Living Network Alliance (DLNA) [2], a forum supported by many companies in the consumer electronics, multimedia, entertainment, and mobile industries.

The key of the UPnP AV architecture and DLNA is multimedia metadata sharing. Therefore, their specifications and guidelines have protocols and constraints for multimedia metadata sharing.

---

[1]DLNA. http://www.dlna.org. A non-profit collaborative trade organization has more than 250 member companies to make their products compatible with each other. The DLNA published its first set of Interoperability Guidelines in June 2004 and the first set of DLNA Certified products began appearing in market thereafter. To become certified, products pass through the DLNA Certification Program.

[2]UPnP AV Architecture. UPnP AV is an audio and video extension of UPnP. It defines three device classes, actions and events among the classes.

2.1 UPnP AV

UPnP was defined by the UPnP Forum [3] as what brings easy-to-use, flexible, standards-based connectivity to peer-to-peer networks. It is an open architecture that uses established standards such as IPv4 [4], HTTP [5], XML [6], and SOAP [7]. The UPnP AV architecture defines AV device classes the general interaction between its Control Points and devices.
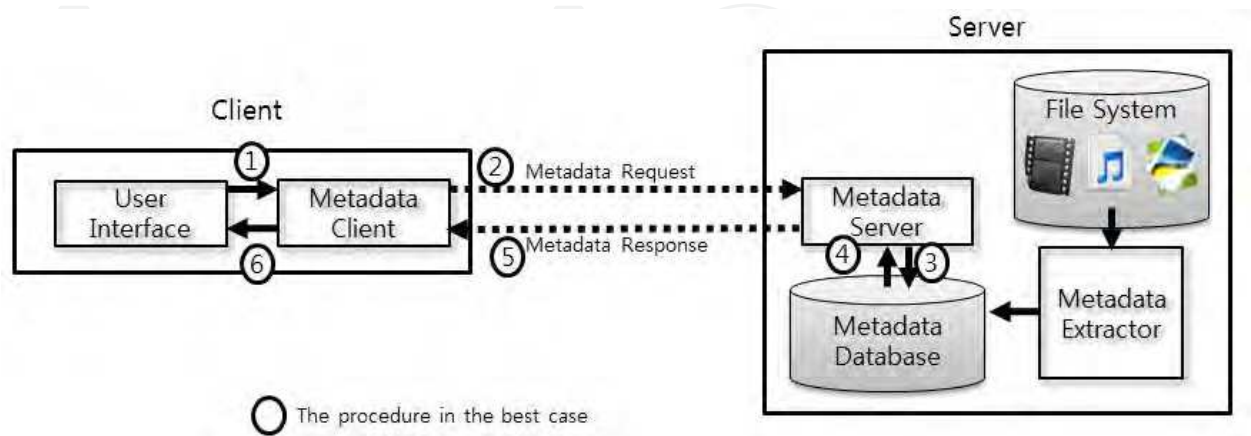

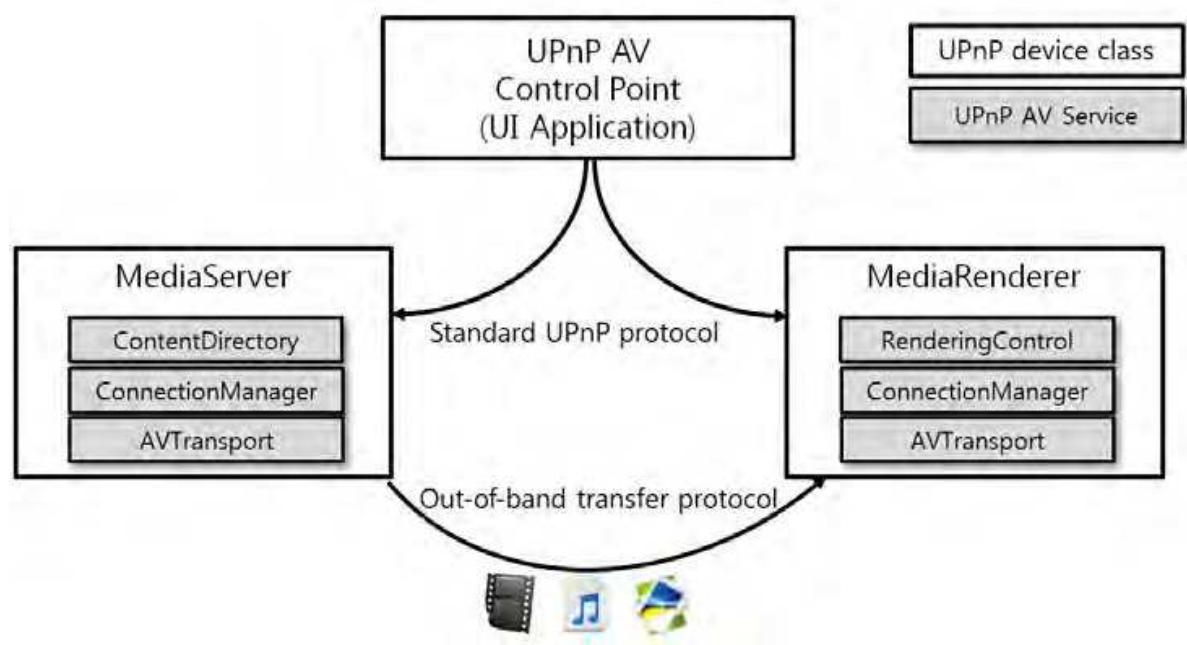Fig. 2. A typical metadata sharing system and the acquisition procedure

Fig. 3. A content playback scenario involves three distinct UPnP AV components: a Media Server, a Media Renderer, and an UPnP AV Control Point

[3]UPnP Forum. http://www. upnp.org. Formed in October 1999, UPnP Forum is an industry initiative of more than 954 leading companies in computing, printing and networking; consumer electronics; home appliances, automation, control and security; and mobile products.
[4]Internal Protocol version 4.
[5]Hypertext Transfer Protocol.
[6]Extensible Markup Language. http:// www.w3.org/XML/.
[7]Simple Object Access Protocol. http://www.w3.org/TR/soap/.

2.1.1 UPnP AV device classes

An UPnP AV Control Point is a device class responsible for coordinating and synchronizing distributed devices and a Media Server [8] is a device class defined as the source device of the media content responsible for exposing and streaming multimedia. In order to correspond with user inputs for the playback of multimedia contents, the Control Point coordinates, synchronizes, and interacts with its devices acting as a source called the Media Server and a sink called the Media Renderer [9].
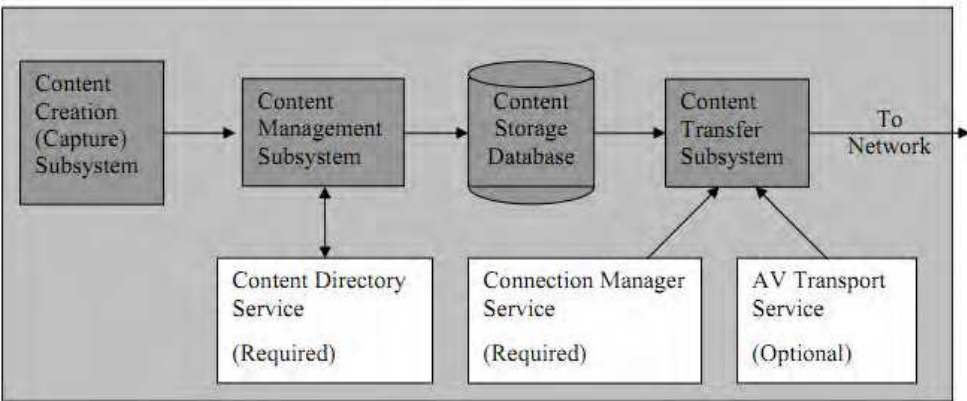
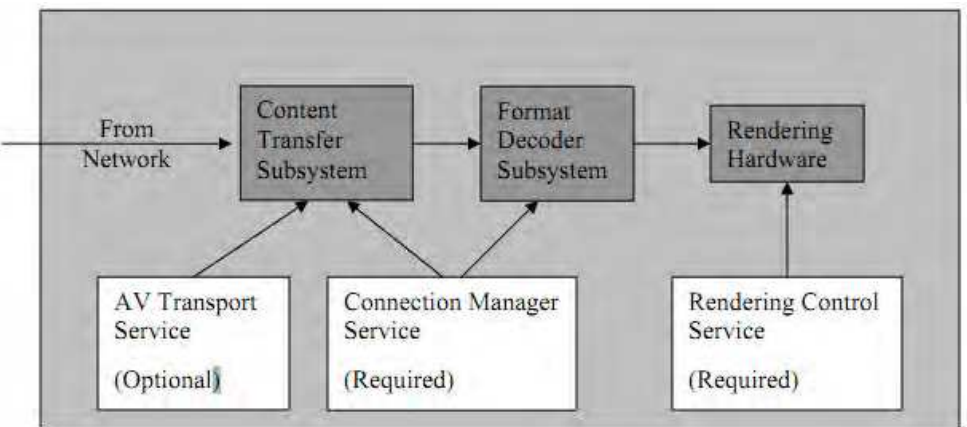Fig. 4. The conceptual structure of UPnP MediaServer

Fig. 5. The conceptual structure of UPnP MediaRenderer

2.1.2 Metadata exchange in UPnP AV devices

The Media Server exposes its contents via the Content Directory service (CDS) which establishes and maintains the hierarchical structure of UPnP container objects. Each object has multimedia content information and metadata such as the type, creator, date, and locator to access it and each CDS may have a unique structure and various objects. UPnP defines two requests to inspect a CDS such as Browse to browse the CDS structure and Search to obtain items conforming to specified search terms. A Browse request is mandatory in all CDSs, but a Search request is optional in UPnP AV specification. UPnP also defines

---

[8]UPnP MediaServer. http://www.upnp.org/specs/av/UPnP-av-MediaServer-v1-Device.pdf.
[9]UPnP MediaRenderer. http://www.upnp.org/specs/av/UPnP-av-MediaRenderer-v1-Device.pdf.

two events such as SystemUpdateID [10] and ContainerUpdateIDs [11] to indicate the change in the CDS. UPnP AV architecture defines the control messages expressed in XML using SOAP. SOAP is similar to RPC where the client sends a request to the server and the server sends a response to the client. This architecture also standardizes the transport layer for SOAP in the architecture using HTTP. SOAP in HTTP is used for high interoperability when exchanging information in networks.
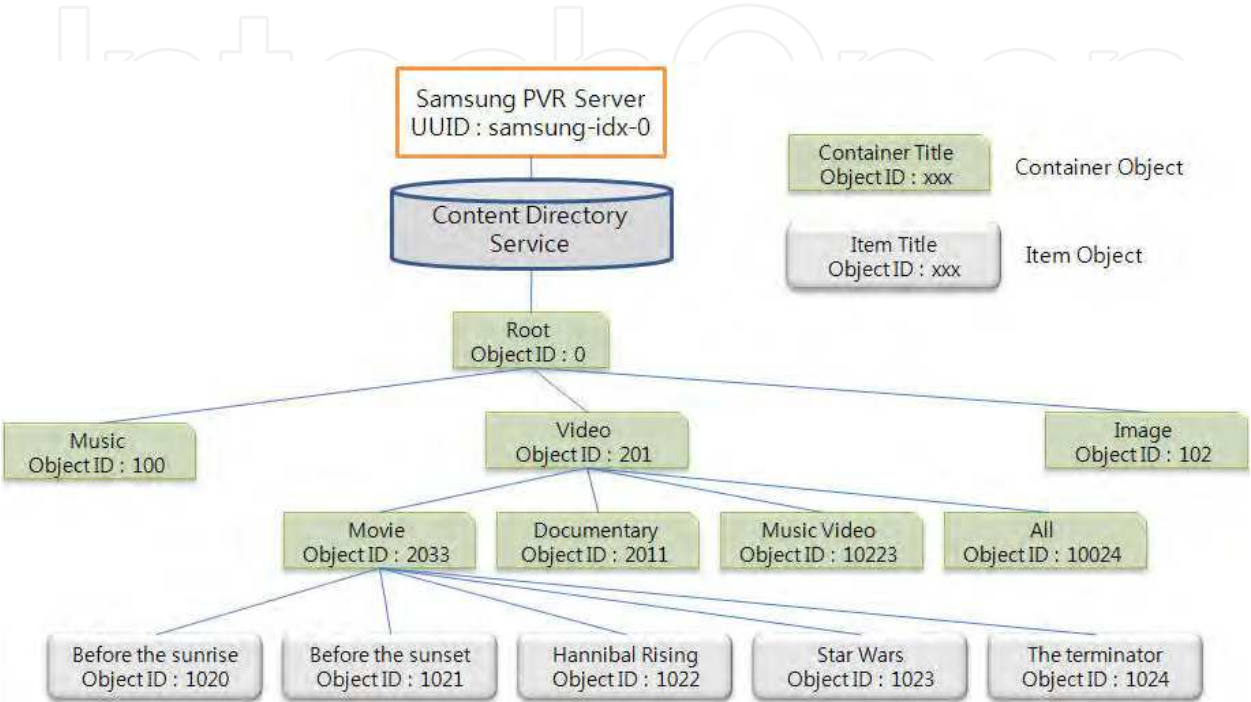

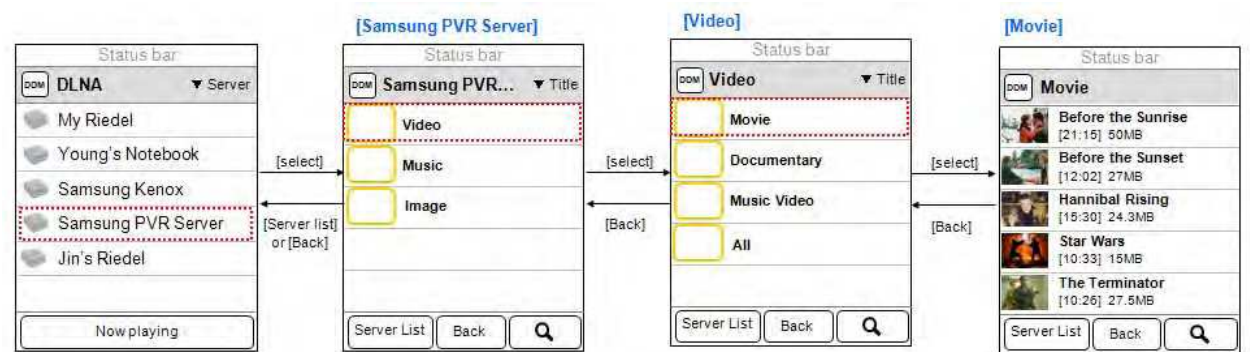Fig. 6. An example of the hierarchical CDS structure

Fig. 7. An example of browsing the content list in the UPnP AV Control Point. The hierarchical structure of the CDS in figure 3 was navigated by Browse requests in this Example

---

[10]SystemUpdateID. When any content in the CDS is added, removed, or changed, the SystemUpdateID event must be delivered to all the subscribed UPnP AV Control Points. A mandatory event of UPnP AV specification.

[11]ContainerUpdateIDs. When any content in specific containers in the CDS is added, removed, or changed, the ContainerUpdateIDs event with the information on the containers should be delivered to all the subscribed UPnP AV Control Points. A optional event of UPnP AV specification.

2.2 DLNA

The aim of the DLNA is to enable end-to-end interoperability among the digital multimedia devices storing, playing and sharing digital content and help put an end to the fragmentation of multimedia sharing standards for multimedia devices. As a collaboration of the world's leading consumer electronics, PC and mobile companies, the DLNA has created design guidelines for DLNA Certified products that can work together — no matter what the brand is. The DLNA has been based on the UPnP AV architecture, and put some constraints on the metadata exchange protocols of the architecture to ensure interoperability among devices in local networks.

Fig. 8. DLNA certified products (http://www.dlna.org/products)

3. Challenges of multimedia metadata sharing solutions

Poor performance due to the round-trip latencies between the servers and clients is a typical problem of network-based solutions. Since some additional protocols to express the metadata hurt the responsiveness of multimedia applications, multimedia sharing systems suffer from serious usability problems due to poor responsiveness. Additionally, because multimedia metadata are distributed across several multimedia devices and the sharing protocols meet some constraints to improve interoperability, the extensibility of the metadata based services is usually poor. For instance, even if you want to sort multimedia contents according to geographic information, the UPnP AV architecture and DLNA don't support that. Thus, you should introduce your own propriety protocol hurting interoperability.

3.1 Performance

A common serious usability problem in multimedia sharing systems in networks is poor responsiveness of metadata acquisition. Generally, 100 to 200ms is the threshold beyond which users will perceive a lag in an application [12]. However, metadata acquisition in the

---

[12]Beigbeder, T., et al., (2003). The effects of loss and latency on user performance in Unreal Tournament, Proc. NetGames, (2004), pp. 144-151.

systems causes long round-trip latencies, due mainly to the protocol exchange mechanism. For this reason, interactive multimedia sharing applications have suffered and caused inconvenience to users. To avoid this problem, some solutions have tried to aggregate all the metadata in a controller device which has a user interface. In that case, a user usually suffers from long warm-up time to gather metadata instead of the responsiveness problem which was discussed above.
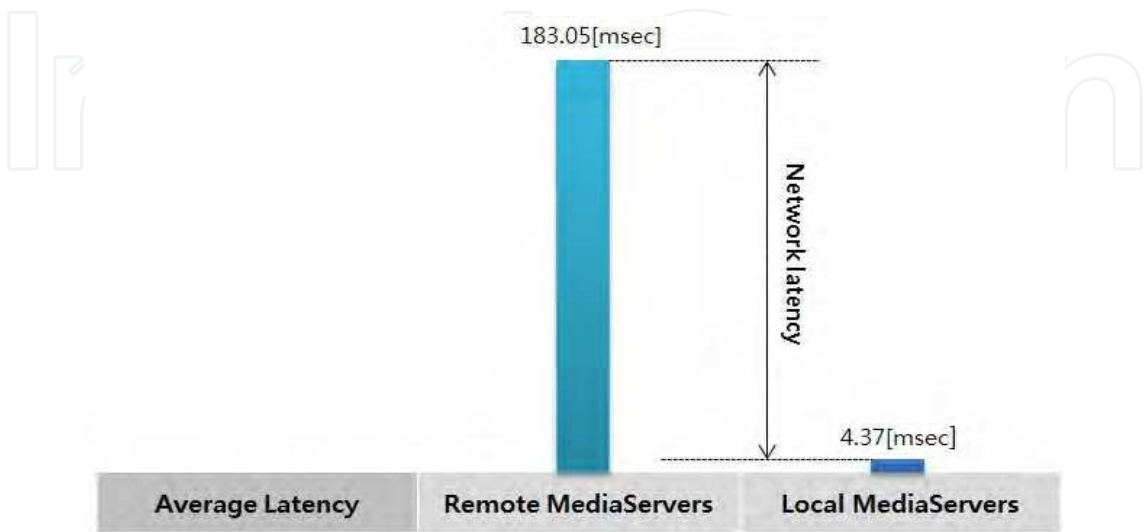
Fig. 9. The comparison of average latencies of UPNP AV's Browse requests to obtain metadata [13]. The round-trip latencies of Browse requests with local MediaServers were definitely short

3.2 Service extensibility problem

Another critical problem is a lack of additional metadata-based functions such as search and sort with a title, date, or type of contents that provide convenient ways for the user to find expected multimedia content. However, if multimedia metadata spreads over several servers, it is difficult to implement these functions. Moreover, the functions also cause high round-trip latencies.

4. Solutions

To address challenges such as poor responsiveness and limitation of service extensibility from a user's perspective as we described above, several solutions such as caching responses, historical and statistical information based prefetching, aggressive link based prefetching, full prefetching, metadata database object acquisition, and the metadata aggregator-based centralized solution have been introduced.

4.1 Caching

Caching the metadata in the client is one classical solution. The response to a user's metadata request is stored in a cache. In this case, since the round-trip latency of a metadata

---

[13]Hochul, S., (2008). Achieving Low Latency of Multimedia Content Browsing in UPnP AV Architectures

request with a cache hit is definitely short, a user may be satisfied with a very fast response. Additionally, with a larger number of cache entries, the cache hit rate tends to be higher.
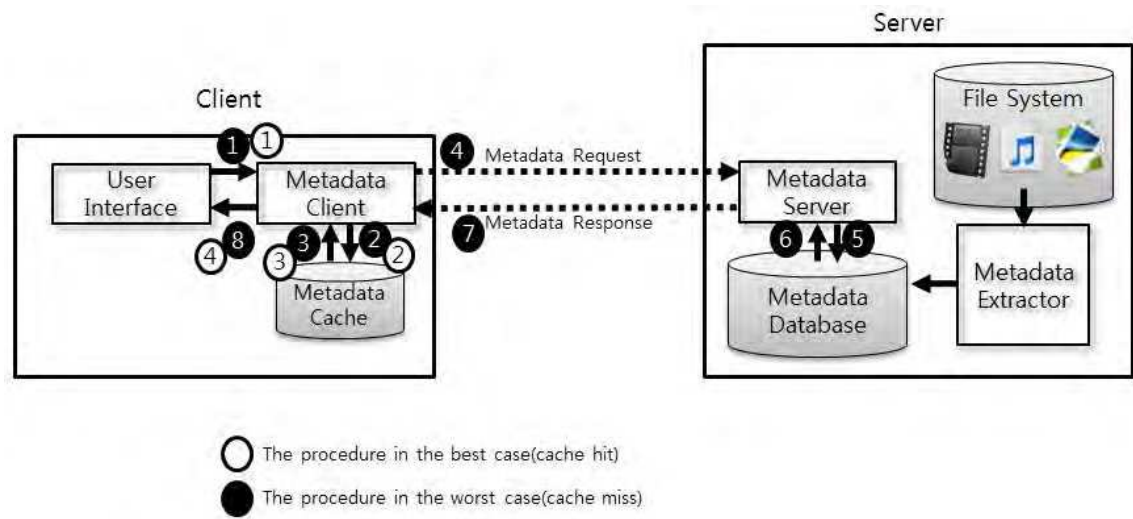

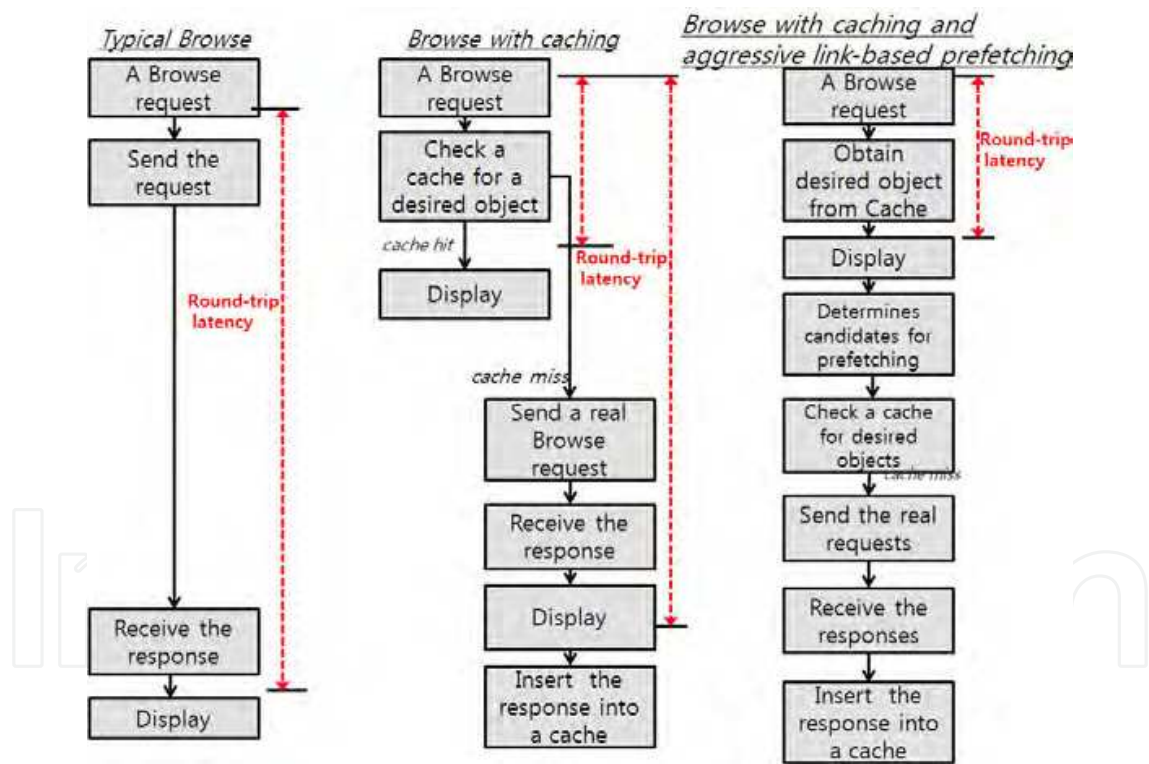Fig. 10. An example of metadata sharing systems with caching and the acquisition procedure

Fig. 11. The comparison of round-trip latency of a user's Browse requests in UPnP AV architecture with the typical, caching, and prefetching

However, cache misses which are known as cold start or compulsory misses during cache warm-up are unavoidable and a cache does not guarantee 100% cache hit rate even after cache warm-up is completed. Moreover, it is difficult for caching to meet users' needs of using a variety of metadata-based functions because the cache can only store part of the whole metadata.
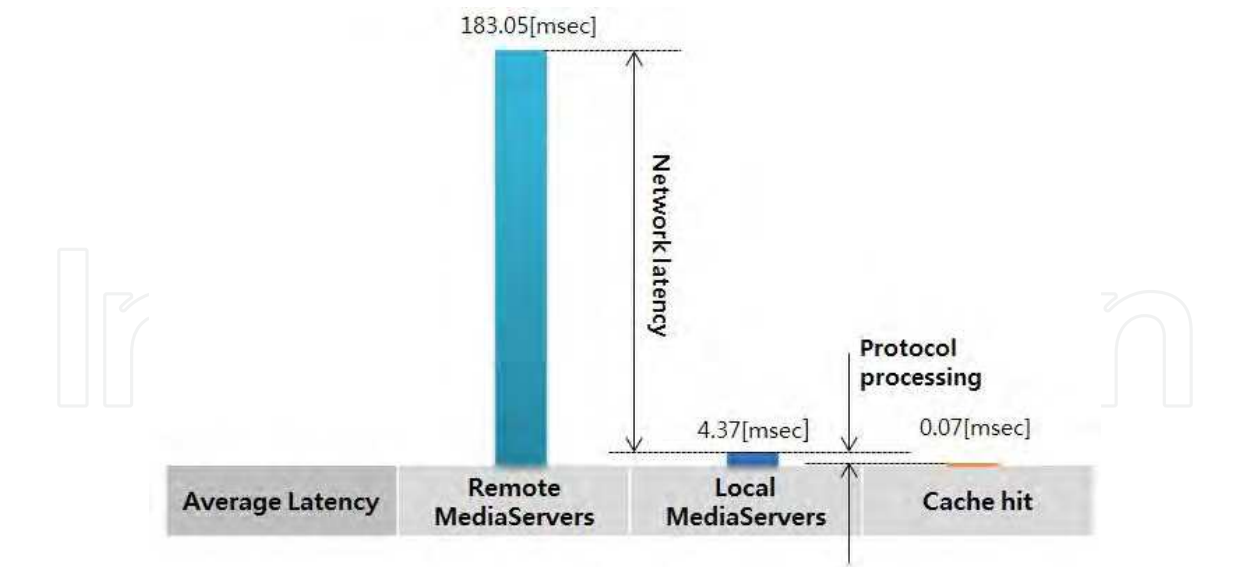
Fig. 12. The comparison of evaluated average latencies of Browse requests. Cache hits reduced the latencies dramatically[14]

4.2 Prefetching

Another solution is prefetching metadata responses to the expected requests. This means that a client sends requests based on users' prior requests and stores the responses in the cache before the user requests them. The prefetching seems necessary to prevent cache misses and this could dramatically improve the round-trip latencies from the user's perspective if the response to the request is already stored in the cache. There are several kinds of prefetching algorithms such as prefetching based on historical and statistical information, aggressive link based prefetching, and full prefetching before a user's first input.
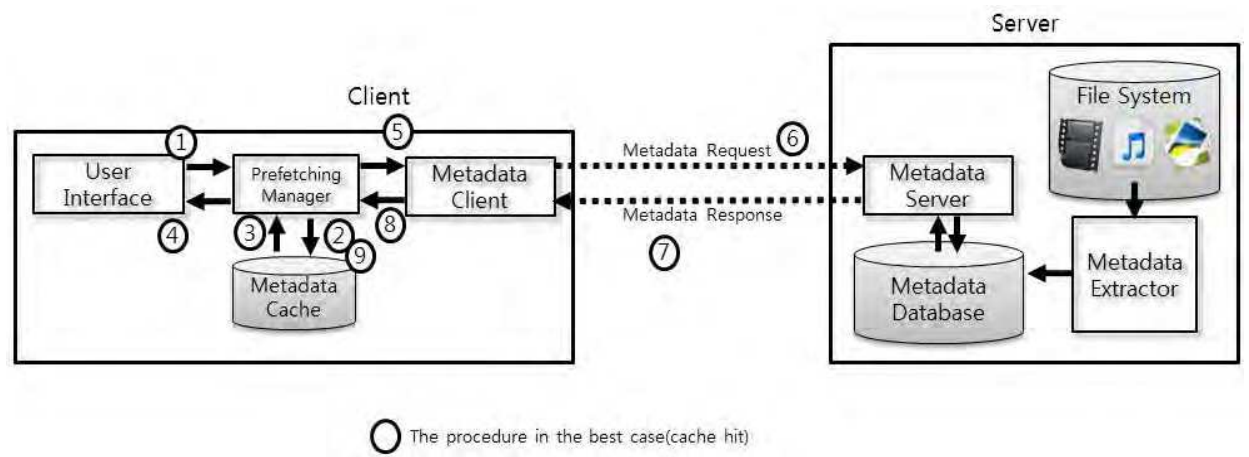

Fig. 13. An example of metadata sharing systems with prefetching. The steps from 5 to 9 are to obtain metadata for users' future requests according to historical and statistical information or the hierarchical structure of the contents before the next request

---

[14]Hochul, S., (2008). Achieving Low Latency of Multimedia Content Browsing in UPnP AV Architectures

4.2.1 Historical and statistical information based prefetching

To increase cache hit rate, we can predict users' expected requests based on historical and statistical data, for example, a prediction based on what kinds of multimedia contents have been chosen or what search keywords have been used by the user. However, cache miss is inevitable because a user's preference can always be changed.

4.2.2 Aggressive link based prefetching

Aggressive link based prefetching means that all possible links that a user may select must be prefetched. Multimedia metadata is usually stored and expressed as a hierarchical structure. So, while a user navigates the structure, all metadata in links the user can move will be obtained in advance before the user moves to one of the links. It ensures a cache hit rate of 100% at all times except in the case that a user's input is faster than the response acquisition for the links. Also, it is not suitable for cases in which a link has a lot of items or that there are a lot of links that the user can move.

4.2.3 Full prefetching

The third solution is more aggressive to improve the usability. The solution is that the entire metadata stored in the servers is copied into the client using metadata acquisition protocols, even if the initial setup time is very long. It establishes the illusion of the original metadata storage at the client. After this has been established, users can access the storage with very low latencies and a variety of metadata-based functions can be supported as if the metadata storage is real. However, the whole metadata acquisition at the initial setup time may cause the huge protocol processing, data access time and the heavy traffic in networks due to very complicated low level protocol exchange, XML processing and retrieval of metadata from the database in the server and to the client.
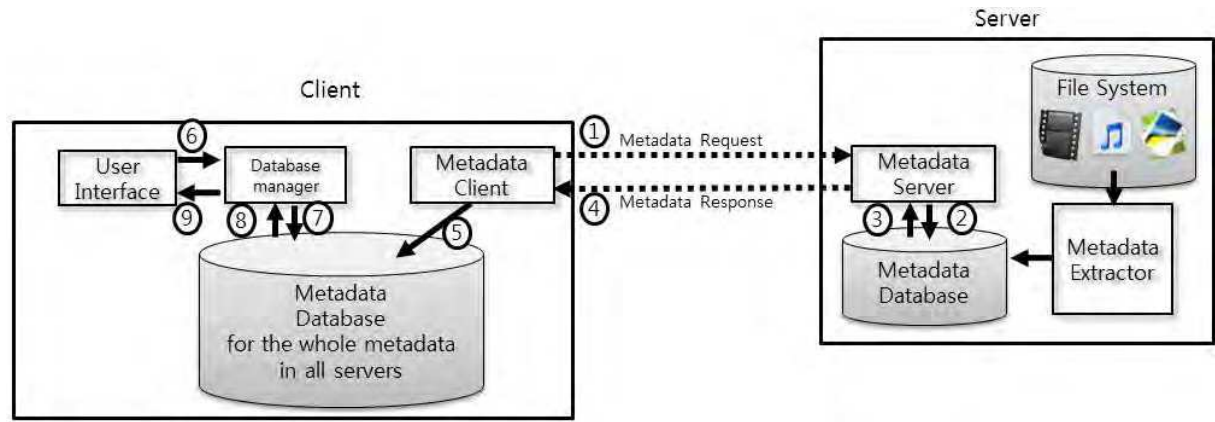
Fig. 14. An example of metadata sharing systems with full prefetching. The steps from 1 to 5 are to obtain the whole metadata in all the servers via metadata exchange protocols and establish the illusion of metadata storage. Then, a user's request refers to the illusion instead of the real servers

4.3 Metadata database object acquisition

The establishment of the illusion of metadata storage at the client is very effective for usability except for a very long initial setup time. Therefore, we propose a more aggressive

solution that copies the metadata database object itself from the servers to the client instead of copying metadata. If a client uses the same schema for metadata database as the servers, we can directly copy the database object from the servers to the client, and the client then can access the database object without a very complicated and redundant metadata acquisition method in the network. Database object acquisition protocol is relatively much simpler than metadata acquisition protocol. For example, a typical multimedia metadata acquisition method uses HTTP and SOAP for extensibility, interoperability, and flexibility of services.

On the other hand, these protocols are very complicated and cause very long round-trip latencies to exchange metadata. However, to obtain the database object, we can use the FTP which is a very simple and fast protocol. Additionally, since multiple database objects can be referred to in order to respond to user's requests, various metadata-based functions can be very easily implemented.

However, this solution requires the unified interface to the metadata database across several servers and clients. Thus, unlike the other solutions, changing both the servers and clients is required to apply this solution to your application.
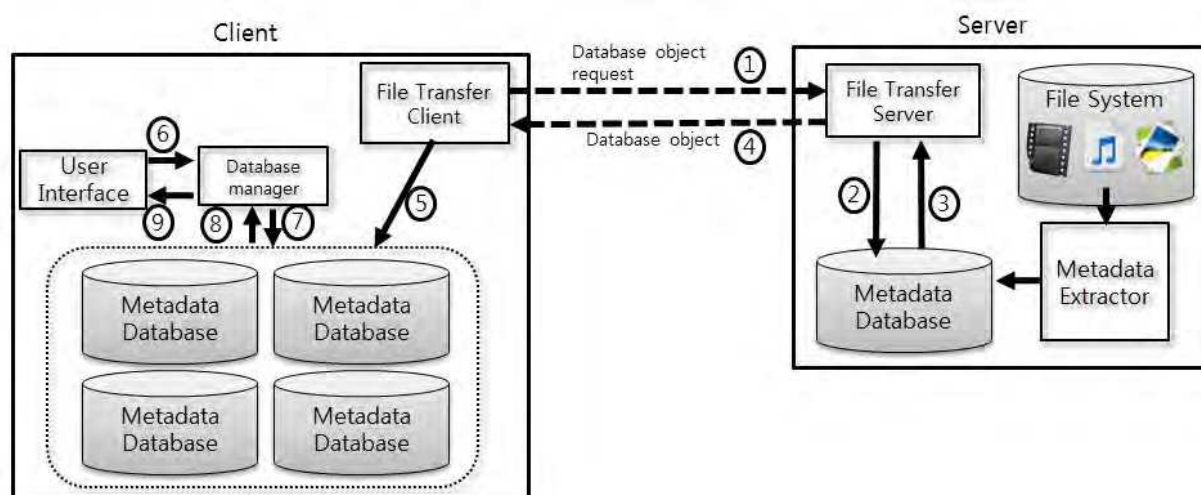

Fig. 15. An example of metadata sharing systems with Metadata database object acquisition. The steps from 1 to 5 are to obtain the whole metadata in all servers via file transfer protocols and establish the illusion of metadata storage. Then, a user's request refers to the illusion instead of the real servers

4.4 Metadata aggregator-based centralized solution

To obtain metadata in distributed systems, a client must access all the servers which causes the service extension problem. For example, if a user wants to get items ordered by dates, the client must send messages to all the servers to obtain metadata, wait for the replies and then sort them. If the whole metadata is aggregated in one server, a client can only access this server instead of all the servers. Also, the caching mechanism or prefetching mechanism can be used for the clients. The aggregator is suitable to run on non-mobile devices working with many metadata servers. If there are a few servers, it is not beneficial to apply this solution to that environment.
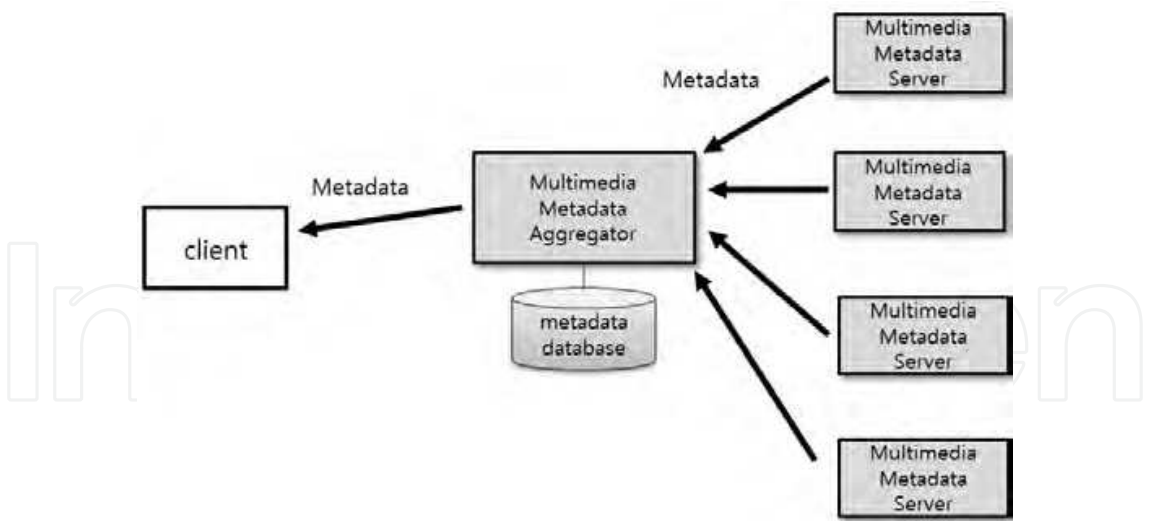
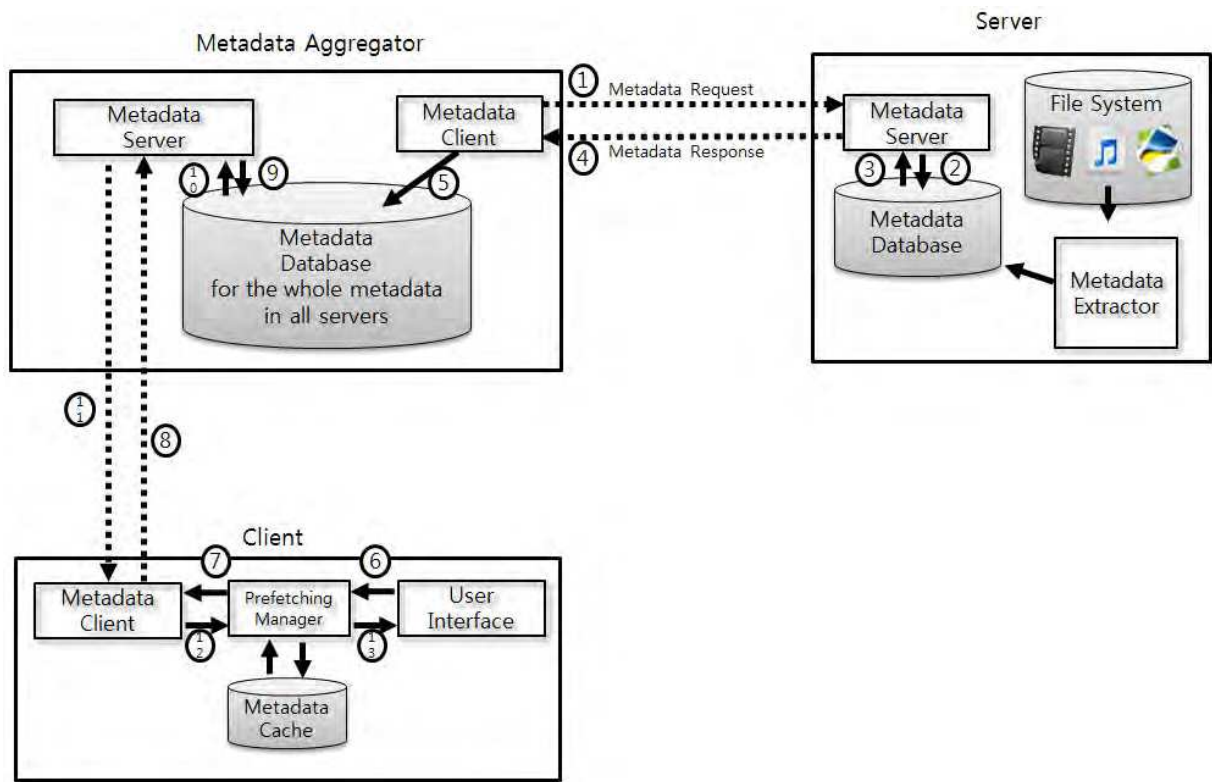Fig. 16. The conceptual diagram of metadata sharing systems with a metadata aggregator

Fig. 17. An example of metadata sharing systems with a metadata aggregator. The steps from 1 to 5 are to obtain the whole metadata from all the servers and fill the metadata aggregator with the metadata from the servers. Then, a client accesses the aggregator instead of all the servers in the network

4.5 Comparison of the solutions

Any solution described above does not resolve both the performance and service extensibility problem of metadata sharing solutions for multimedia sharing systems as follows.

Metadata sharing solutions	Required time for the first service	The best case round trip latency	The worst case round trip latency	Possibility of the worst case	The extensibility of metadata based services	The scope of implementation
Typical	No	Long	-	-	Low	-
Historical and statistical information based prefetching	No	Short	Long [15]	High [16]	Low	Client
Aggressive link based prefetching	No	Short	Long	Low [17]	Low	Client
Full prefetching	Very long	Short	-	-	High	Client
Metadata database object acquisition	Long	Short	-	-	High	Server and Client [18]
Metadata aggregator (client)	No [19]	-	-	-	High	Aggregator and Client

Table 1. Comparison of the metadata sharing solutions

5. Conclusion

In this chapter, we have presented the typical problems of metadata sharing. The latency problem and service extensibility problem are typical and crucial in industrial solutions but very difficult to resolve. To address those problems, caching responses, historical and statistical information based prefetching, aggressive link based prefetching, full prefetching, metadata database object acquisition, and the metadata aggregator-based centralized solution have been introduced and these have their own limitations, pros, and cons. Thus, you should understand that each solution has its own pros, cons, and limitations and figure out what solutions can be applied to your application. Most of all, the most important factor you should consider when you choose the solutions is how to satisfy users of your application.

---

[15] It depends on the cache algorithm, the number of cache entries, and cache replacement algorithm.

[16] When cache-miss happens, long round-trip time is required.

[17] It depends on the number of links, the duration between a user's two inputs, and the speed of servers' response.

[18] The metadata database object acquistion protocol must be implemented on the servers and clients. Also, the clients must be able to figure out that the servers support that protocol.

[19] It takes long time to the aggregator gathers the whole metadata from the servers in the network. Then the clients can use the metadata immediately.

6. References

Andresen, D., et al., (2004). *LYE: a high-performance caching SOAP implementation*, Parallel Processing, ICPP 2004. International Conference, pp. 143 - 150 vol.1.

Beigbeder, T., et al., (2003). T*he effects of loss and latency on user performance in Unreal Tournament*, Proc. NetGames, (2004), pp. 144-151.

Berners-Lee, T., et al., (1996). *Hypertext Transfer Protocol – HTTP/1.0*, RFC 1945

Cao, P., Felten, et al., (1995). *A study of integrated prefetching and caching strategies*, Proceedings of the 1995 ACM SIGMETRICS joint international conference on Measurement and modeling of computer systems, (May 1995), pp. 188-197, Ottawa, Ontario, Canada.

Cohen, E., & Kaplan, H., (2000). *Prefetching the means for document transfer: A new approach for reducing Web latency*. In Proceedings of IEEE INFOCOM, Tel Aviv, Israel, (March 2000)

Davis, D., Parashar, P. (2002). *Latency Performance of SOAP Implementations Cluster Computing and the Grid*, 2002. 2nd IEEE/ACM International Symposium (May 2002) pp. 407–407.

Davison, B., (2002). *Predicting web actions from HTML content*, Proceedings of the thirteenth ACM conference on Hypertext and hypermedia, (June 2002), College Park, Maryland, USA.

Devaram, K & Andresen, D., (2003). *Soap optimization via parameterized client-side caching*. Fifteenth IASTED International Conference on Parallel and Distributed Computing and Systems, Marina del Ray, CA, USA, (Nov. 2003), pp. 785-790.

Duchamp, D., (1999). *Prefetching hyperlinks*, Proceedings of the 2nd conference on USENIX Symposium on Internet Technologies and Systems, (October 1999), pp. 12-12, Boulder, Colorado.

Duncan, B, & Heron, R., (2005). *Query caching in a system with a content directory service*, Publication Number:WO/2005/031607.

Hochul, S., (2008). *Achieving Low Latency of Multimedia Content Browsing in UPnP AV Architectures*, The Second International Conference on Mobile Ubiquitous Computing Systems Services and Technologies, pp. 56-63.

Kroeger, M., et al., (1997). *Exploring the Bounds of Web Latency Reduction from Caching and Prefetching*, Proceedings of the USENIX Symposium on Internet Technologies and Systems, (December 1997)

Padmanabhan, N., & Mogul, C., (1996). *Using predictive prefetching to improve World Wide Web latency*, Computer Communication Review, (July 1996), pp. 22-36, Proceedings of SIGCOMM '96.

Ritchie, T. & Kuehnel, T., (2002). *UPnP AV Architecture: 1*, UPnP Forum, Available from http://www.upnp.org/specs/av/UPnP-av-AVArchitecture-v1-20020625.pdf

Ritchie, T. et al., (2003). *UPnP Device Architecture 1.1*, UPnP Forum,   Available from http://www.upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v1.1.pdf

Sarukkai , R., (2000). *Link prediction and path analysis using Markov chains*, Proceedings of the 9th international World Wide Web conference on Computer networks : the international journal of computer and telecommunications networking, (June 2000), pp.377-386, Amsterdam, The Netherlands.

Multimedia - A Multidisciplinary Approach to Complex Issues
Edited by Dr. Ioannis Karydis

The nowadays ubiquitous and effortless digital data capture and processing capabilities offered by the majority of devices, lead to an unprecedented penetration of multimedia content in our everyday life. To make the most of this phenomenon, the rapidly increasing volume and usage of digitised content requires constant re-evaluation and adaptation of multimedia methodologies, in order to meet the relentless change of requirements from both the user and system perspectives. Advances in Multimedia provides readers with an overview of the ever-growing field of multimedia by bringing together various research studies and surveys from different subfields that point out such important aspects. Some of the main topics that this book deals with include: multimedia management in peer-to-peer structures & wireless networks, security characteristics in multimedia, semantic gap bridging for multimedia content and novel multimedia applications.

How to reference
In order to correctly reference this scholarly work, feel free to copy and paste the following:

INTECH
open science | open minds