RC23617 (C0505-006) May 25, 2005
Computer Science

IBM Research Report

Capability and Context Aware Content Indexing in P2P Network

Lei Ma, Yu Chen Zhou
IBM Research Division
China Research Laboratory
HaoHai Building, No. 7, 5th Street
ShangDi, Beijing 100085
China

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

Capability and Context Aware Content Indexing In P2P Network

Lei Ma, Yu Chen Zhou
IBM China Software Development Laboratory
{malei, zhouyuc}@cn.ibm.com

Abstract

Peer-to-peer (P2P) networks have become an important infrastructure and been involved in more sophisticated system like secure distributed content management. P2P content indexing is one of the key features of such system, and is based on centralized facilities conventionally. In this paper we present a novel approach for capability aware dynamic indexing for purely P2P network, especially private network. Leveraging RDF, peer could dynamically retrieve content information based on composite capabilities of local device. Based on nature of fully connected topology, simple distributed transactions in both active and passive modes implement adaptability for the dynamic changes of network connection, sharable content, capabilities and status of peers. We show how this approach can enhance the context sensitive indexing functionality and meet the requirements in a dynamically changed purely P2P network.

Keywords: Content Indexing, Peer-to-peer, RDF, Device Capability.

1 Introduction

Peer-to-peer network are distributed network that consists of interconnected nodes able to self-organize into network topologies with the purpose of sharing resources and without any centralized control or hierarchical organization. Comparing with the client/server network, one of its characteristics is to allow local resources to be shared directly without the need for intermediate servers.

With more and more network technologies applied in the private network, such as a home network, the requirement is arisen to share and protect content effectively within the P2P environment. For example, a home network needs to share diversified data, from plain text data to multimedia data, among various devices with different capabilities, such as PCs, the residential gateway, the home media center and mobile devices. A lot of efforts have been made to handle the content information shared within the P2P networks. Conventional solution is based on client-server mode, in which content information is stored in a centralized server and clients get the information from the server. Such solutions use technologies derived from enterprise technologies such as JNDI/LDAP. And some dedicated solutions are defined for home network multimedia applications, for example, the registration service of the open source project Network-Integrated Multimedia Middleware [1]. While such solutions are difficult to solve the problems caused by the nature of the peer-to-peer network, for it is hardly to have a persistent and high capable centralized server in the home network, and the fault of the server device may caused a fatal problem to content indexing of the whole network.

In order to support advanced applications, some capability information is used in mobile web access. The typical ones are W3C Device Independent and CC/PP [2], and also OMA UAProf [3]. For these technologies mainly focus on conventional client-server architecture, they also can not meet the need of P2P networks.

In this paper, we present an approach for content indexing in the P2P private network, in which peer provides its capabilities to other devices to get suitable content information. This approach provides flexible content indexing function for Content Protection for Workplace Client Technology (CPWCT), which is a broadcast encryption [4], based content sharing and protection solution for P2P private network, and could also be widely used in more applications. It relies on the W3C metadata standards RDF [5, 6, 7] to describe device capabilities and meets the characteristics of peer-to-peer private network. Beside these, it handles the synchronization of content information and fault tolerant.

In this approach, peers retrieve content information based on its media capability, system capability and even user profile, which reflect the device characteristics and the user preference. Each content holder provides the suitable content information to requesting devices according to their individual capabilities. The process can provide an accurate, flexible and on demand content indexing way and get rid of the risk that a device wants to render content out of the scope of its physical capability. It realizes the capability and content aware content indexing. Based on nature of fully connected topology, simplified distributed transactions in both active and passive modes implement adaptability for the dynamic changes of network connection, sharable content, capabilities and status of peers

The rest of this paper is structured as follows. Section 2 introduces the architecture and strategy on CPWCT content indexing. Section 3 defines the RDF based capability presentation. Section 4 describes content awareness of the CPWCT content indexing. Section 5 presents the content indexing transactions that implements context awareness. Finally, we summarize the approach and introduce the future work in Section 6.

2 Architecture and Strategy

The CPWCT provides automatic device discovery, inter-cluster authorization, clustering and secure data transportation mechanism based on broadcast encryption technology. A group of CPWCT devices forms a trusted community eXtensible Content Protection (xCP) [8, 9, 10] cluster, which defines the social context in which the devices interact. The behavior of CPWCT is a kind of social behavior including both collaboration and competence. The CPWCT devices collaborate in tasks for including clustering, content dissemination and indexing. CPWCT devices actively contact with others to join the xCP cluster and share local content to trusted devices, in the mean time to authorize devices and catch content from other devices. An xCP cluster could be abstracted as a set *Cluster={D, S},* where D is trusted device set in the xCP cluster, and S is the cluster context. The structure of CPWCT is shown in Figure 1.

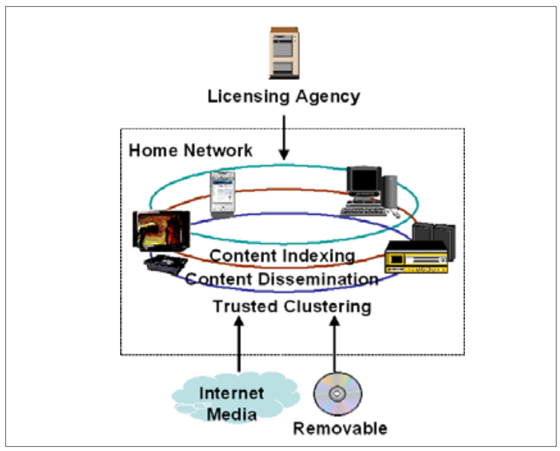

Figure 1 CPWCT System in Home Network

In the content indexing level, the key points are devices' capabilities and interactions devices. We considered the content indexing approach from the following aspects: (1) the description to the static objects in the system, including devices, (2) the description of distributed transactions for content indexing in the system, including the devices' inter-activities in content indexing level.

For an xCP cluster, the global snapshot is

$D=\cup d_i$, $B_{cs}=\cup B_{s,i}$, $B_{s,i}=\{cf_1, \ldots cf_m\}$, where D is the set of devices of the xCP cluster, d_i is the devices in the cluster, B_{cs} is the set of all xCP content files in the cluster, $B_{s,i}$ is the set of xCP content files managed by a_i. The global content information snapshot retrieved by a_i is $CI_i=\cup f(B_{s,i},C_i)$ and $j\neq i$, where C_i is the capability of a_i and CI_i is only a subset of B_{cs}. The basic assumption is $B_{s,i}=\varphi$, if the network connection to d_i is disconnected, d_i is shutdown or d_i has no local xCP content file.

In this static view, a device a_i could be described from abstract aspects of Media Capability, System Capability and User Profile shown in Figure 2.

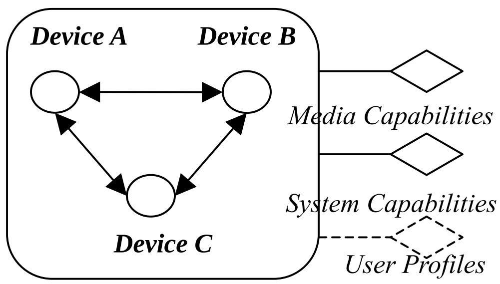

Figure 2. Devices Abstract View

Media Capability - It describes media types accepted by a device. It describes the media features of a device. For a device, with different kinds of software and hardware, the device's media capabilities are various. A PDA may have windows media play, so it can support window media video/audio media. A digital TV may integrate a hardware MPEG 4 decoder, so it can play media in MPEG 4 format. When sharing content information in the private cluster, a device can provide its media capability to other devices to request the media fitting for it.

System Capability - System capability describes physical characteristics of the device. It contains the descriptions of the device, such as CPU speed, memory size, and screen size and device type. This information provides a reference to decide which kinds of media can be share to the device. For example, the PDA with the limited hardware resource could only play small size of media or the streaming media, but a home media center doesn't have such a limitation and can be shared with large and high quality media contents.

User profile - User profile describes the user preference and maybe other user's information. It contains the description of the user id and his/her preference. Different users may have different preferences on different kinds of device. For example, the user may prefer to listen to mp3 in a PDA when he is cooking and may prefer to watch a high quality Hollywood movie in his home media center. All this user profiles will be record by devices in the cluster. And based on a certain statistical strategy, devices can provide different services to users.

3 RDF Based Capability Presentation

In order to implement capability and content aware content indexing in a peer-to-peer network, we must find out a way to describe the device's capability. With the description, devices can exchange their capabilities without the explicit users' participation.

RDF is used to annotate resources on the Web and provide the means by which computer systems can exchange and comprehend data. All resources are identifiable by unique resource identifiers. Annotations about resources are based on various schemas that are defined in RDFS and are stored in RDF repositories.

Using RDFS, we can represent schemas based on device, capability and value, to define the vocabulary used for describing device capabilities. RDF triple <device, capability, value> represents specific annotations, where device identifies the device, capability specifies what properties the device has, and value specifies the value of those properties.

Using the formal model of RDF, we can describe the media capability and system capability as below.

When describing media capabilities, the triple of $<d_i, MC_i, MT_i>$ ($d_i \in D$) is defined, where d_i represents device i, D is the set of devices in the cluster, MC_i represents media capabilities of device i and MT_i represents media types of device *i*. We can view a set of statements as a directed labeled graph model like Figure 3, which represents

The MC_i of d_i are *wmv, rm and mov*.

In the statement, *wmv*, *rm* and *mov* represent different media types.

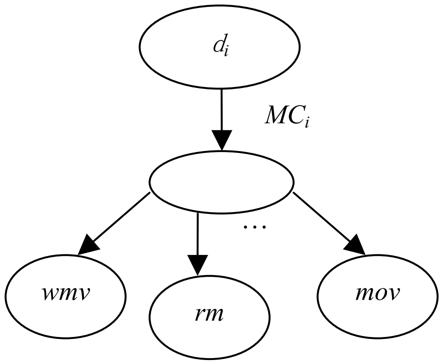

Figure 3. Media Capability Model

When describing system capability, we can use the one of the container objects defined by RDF to present the list of resources or literals likes Figure 4, which represents

The d_i has the MS_i is *32M* and CPU_i *speed* is *1.5G Hz*. The id_j is assigned to d_i.

In the statement, $d_i \in D$, MS_i represents memory size of device *i* and CPU_i represents CUP speed of the devices *i*.

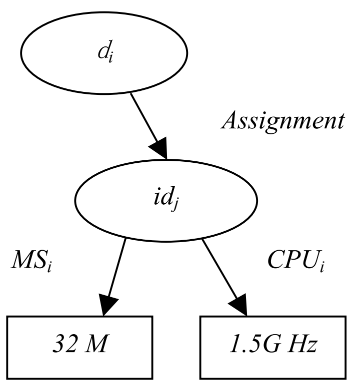

Figure 4. System Capability Model

Here we give out a representation of device capabilities based on RDF/XML format in Figure 5. The representation contains the media capabilities and system capabilities. It is used in indexing transactions to provide the conditions for

```
<rdf:RDF>
 <rdf:Description about="http://www.ibm.com/di ">
  <m: MCi>
   <rdf:Bag>
    <rdf:li type="wmv"/>
    <rdf:li type="rm"/>
    <rdf:li type="mov"/>
   </rdf:Bag>
  </m: MCi >
  <s:Assignment>
   <rdf:Description about="http:// www.ibm.com/idj">
    <v: MSi>32M </v: MSi>
    <v: CPUi>1.5G Hz </v: CPUi>
   </rdf:Description>
  </s: Assignment >
 </rdf:Description>
</rdf:RDF>
```

Figure 5. Device Capability in RDF Format

over specific responsibilities. Because of these characteristics, the CPWCT network topology gets rid of the dependence to super peers. Devices can detect the changes of devices in the cluster using CPWCT infrastructure and then use the RDF-based device capability description to request the new proper content information in the cluster

The network topology change in the different cases:

(1) When a new device joins a cluster, the CPWCT infrastructure will detect the change and notify other devices in the cluster to request content information from the new device. This means a new device will receive N (N is the number of devices in the old cluster) requests message in the cluster. After this, the new device will return the suitable content information to each of N devices. As for the new joined device, it will send N request message to the devices in the cluster to request the content information, which is suitable for itself, and it will receive N response message. All together, when a new device joins a cluster, there will be 4N messages to be transmitted to let the devices get their on demand content information.

(2) When a device leaves from a cluster, the CPWCT infrastructure will detect the change and notify other devices in the cluster to remove content information came from that left device. And this process doesn't need messages exchange between devices in content indexing level.

There are two advantages of adopting pure peer-to-peer topology in content indexing. Firstly, the pure peer-to-peer network meets the characteristic of the private network, especially home network, in which any device could join and leave the network at any time. Moreover, the request/response sessions among each device bring the possibility of capability and content aware content indexing, for each request/response session could contain querying information and result based on each device capabilities. This brings the dynamic characteristic to the content indexing.Secondly, the pure peer-to-peer network allows the devices to be highly independent autonomous devices. In the network, each device holds one suitable content information list. Meanwhile, the CPWCT infrastructure will provide the reliable change information of the cluster. So each device will not depend on any other device to maintain the content information in the cluster. Devices are not only a content information provider but a consumer. This brings the independent characteristic to the content indexing.

4.2 Synchronization and Fault Tolerance

In this approach, each device in the cluster maintains a table of the whole content information in the cluster. And the device will change its content information table in following cases (1) when a device is notified that the change of cluster (device joins or exits) happens, it will modify its table. (2) When a device receives a refresh message and gets response, it will change the table. (3) When a device is notified its local contents changed, it will change the table and send refresh message to other devices. Only through these three entries can a device modify its table, and because the events are atomic events, which will happen and execute one after another in programming level, this could ensure the content information in the cluster is synchronized.

As for the fault tolerant, when a device crash and exit to the cluster without notifying others, the infrastructure of CPWCT could detect the change and notify other devices to delete the content information from this device. A device failed to receive correct responses from another device in following cases: (1) the device is disabled, crashed or turned off; (2) a transport problem occurs in the network. If there isn't any response in the waiting time, the requesting device will consider that the requested device has exited the cluster, and it will delete all the content information from that "dead" device and ignore those responses out of the waiting time.

5. Distributed Indexing Transactions

Because we adopt the pure peer-to-peer network topology, one of the difficulties in an xCP cluster is that no device has access to the global state of the system; especially the devices are loosely coupled. In this approach, the content information could be updated in two modes: active and passive modes. And a simple message set is designed to define the interaction among CPWCT devices for content indexing.

The message set is $M=\{m_{req}, m_{res}, m_{ref}\}$, where m_{req} is message sent to request shared content information from other devices, which contains the media capability, system capability and even user profile of the requesting device in RDF format; m_{res} is response message which contains local content information filtered according to the requesting device capabilities; m_{ref} is notification message which is sent to notify other CPWCT device of local content change.

Based on these three messages, two kinds of operation model could happen in the content indexing.

Active Operation - When a device joined the cluster, it requests the content information in the cluster. The active operation is shown in Figure 6. Device A joins the cluster consisting of device B and C, it sends m_{req} to all the other devices in the cluster. The m_{req} contains its device capabilities. After receiving m_{res} from B and C, A will use the returned data profiles to show all suitable content

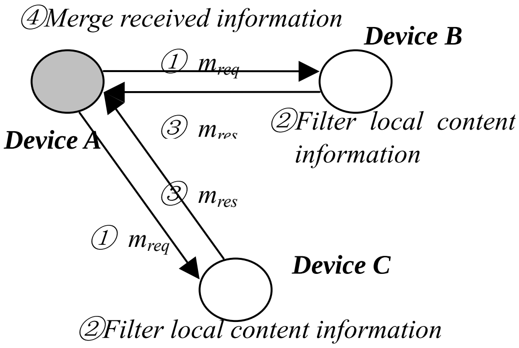

Figure 6. Active Operation

and their information to the user. Then, the user can manipulate these sharing data without carrying whether they can be play in A.

When devices, such as B and C in Figure 6, receive m_{req} from A, they will parse the messages and get A's capabilities. Using A's information, B and C could find out proper content for A. For example, if A is a PDA and the user likes to listen to music in it, then the information of music content in B and C will be found out.

After filtering their local data profile, B and C will send m_{res} containing the matching content information to A. When receiving m_{res} from other B and C. A merges all received information together. Content's name, type and location add together as its key index. As far as the merge completes, A can provide the whole suitable content to the user.

Passive Operation - When the content of a device changed, the device requests the other devices in the cluster to refresh their content

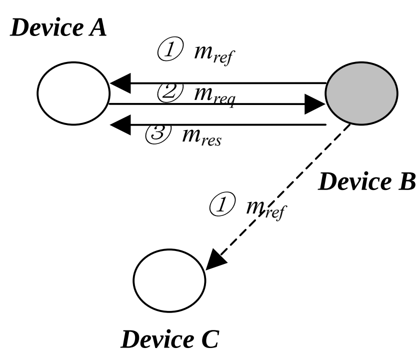

Figure 7. Passive Operation

information. The passive operation is shown in Figure 7.

When the content of device B changed, B sends m_{ref} to notify A and C to request new data profile from B. A and C will send m_{req} to B, and the active operation will be repeated.

For each CPWCT device, a event set represented by $E = \{e_{mc}, e_{sc}, e_{lc}\ e_{as}, e_{cl}, e_{req}, e_{res}, e_{ref}\}$ is defined to trigger actions for content indexing, where e_{mc} and e_{sc} are events generated when local media capabilities change (e.g., media player is added or removed) and system capability change (e.g., memory, hard disk or network bandwidth is changed); CPWCT device generates e_{as} when its status changes, typically, when the device is started; e_{lc} is used to notify changes of local content, (e.g., content file is ingested or removed); e_{cl} is event generated when cluster topology is changed (e.g., a device join or leave the cluster); e_{ras} is event generated when local device is disconnected from other device, it could be caused by either by disconnection of network or remote device shutdown; e_{ref} and e_{res} are generated when m_{req}, m_{res} and m_{re} messages are received. Following are the actions triggered by the events.

$$requestall : \left\{ \begin{array}{l} e_{nc} = \text{CONN} \\ e_{as} = \text{START} \\ e_{ref} \end{array} \right\} \longrightarrow S(m_{req}, Cluster)$$

$$request : \left\{ \begin{array}{l} e_{cl} = \text{JOIN} \\ e_{ras} = \text{START} \\ e_{cl} = \text{JOIN} \\ e_{ref} \end{array} \right\} \longrightarrow S(m_{req}, a_i)$$

$$notifiyall : \left\{ \begin{array}{l} e_{lc} \\ e_{mc} \\ e_{sc} \end{array} \right\} \longrightarrow S(m_{req}, a_i)$$

$$response : e_{req} \longrightarrow S(m_{req}, a_i)$$

6 Conclusion and Future Works

In this paper we present a novel approach for capability aware dynamic indexing for purely P2P network, especially private network. Leveraging RDF, peers in such system could dynamically retrieve content information based on composite capabilities of local device. RDF used in peer-to-peer network allows a more complex and flexible device description. This significantly reduces the useless information exchanged in the network and make device dynamically adjust content information according to the devices changes in peer-to-peer network. Based on nature of fully connected peer-to-peer topology, simplified distributed transactions in both active and passive modes provide adaptability for the dynamic changes of network connection, sharable content, capabilities and status of peers. We show how this approach can enhance the context sensitive indexing functionality and fault tolerance in a dynamically changed purely peer-to-peer network.

In current implementation we only involved in media capability and system capability. With the requirement of consideration of user preference, user profile will become more and more important in content indexing. Actually through adding the user profile abstract, the device can serve as an agent, which can record the users' preferences and based on the preferences provides different content indexing result to different users. For example, if the user likes to listen to music in a PDA, the PDA will set a high priority to music content when content indexing and lists the music in the front of the media list; And if the user prefers, fiction movie, when he open digital TV, the TV set a high priority to fiction movie when content indexing and lists the fiction movies in the front of the media list. The future work could base on the prior research of intelligent agent to figure out a suitable algorithm for peer-to-peer network.

About the Authors

Lei Ma is a software engineer at IBM China Software Development Laboratory. His research interests include pervasive computing and distributed system. Ma received a M.S. in computer science from Northwestern Polytechnical University, China.

Yu Chen Zhou is an advisory software engineer at IBM China Software Development Laboratory. His research interests include pervasive computing, distributed system and multimedia communication. At present his work focuses on PvC device software architecture. Zhou received a Ph.D. in computer science from Beijing University of Aeronautics and Astronautics. He is a member of the IEEE and the ACM.

Reference

[1] Computer Graphics Lab, Saarland University, Saarbrücken, Germany, "The Network-Integrated Multimedia Middleware (NMM): Basic Introduction". http://graphics.cs.uni-sb.de/NMM/-current/Docs/intro/index.html

[2] W3C. "CC/PP Information Page". http://www.w3.org/Mobile/CCPP/.

[3] Open Mobile Alliance. "OMA-UAProf-v2_0-20030520-C". 2003. http://www.openmobilealliance.o.rg/release_program/docs/UAProf/OMA-UAProf-V2_0-20030520-C.PDF

[4] Jeffrey B. Lotspiech, Stefan Nusser, Florian Pestoni. "Broadcast Encryption's Bright Future". Computer magazine (Vol. 35, No. 8), August 2002

[5] W3C. "Resource Description Framework (RDF) Model and Syntax Specifation". Recommendation REC-rdf-syntax-19990222, February 1999.

[6] Lassila, O., Swick R. "W3C Resource Description Framework model and syntax specification", Feb. 1999. http://www.w3.org

[7] Brickley, D., Guha, R. V. "RDF vocabulary description language 1.0: RDF Schema". http://www.w3.org/

[8] Jeffrey Lotspiech, Stefan Nusser, Florian Pestoni. "Anonymous Trust: Digital Rights Management Using Broadcast Encryption". Proceedings of The IEEE, Vol. 92, NO. 6, June, 2004.

[9] Florian Pestoni, Jeffrey B. Lotspiech, Stefan Nusser. "xCP: Peer-to-Peer Content Protection". IEEE Signal Processing Magazine Magzine, March 2004.

[10] Jeffrey B. Lotspiech, Dalit Naor, Florian Pestoni. "Secure Local Agreement in a Peer-to-Peer network"

[11] Wolfgang Nejdl, Martin Wolpers, Wolf Siberski, Christoph Schmitz, Mario Schlosser,

Ingo Brunkhorst_ , Alexander Loser. "SuperPeer Based Routing and Clustering Strategies for RDFBased PeerToPeer Networks".

[12] Wolfgang Nejdl, Wolf Siberski and Michael Sintek. "Design Issues and Challenges for RDF- and Schema-Based Peer-to-Peer Systems". SIGMOD Record, Vol. 32, No. 3, September 2003.

[13] Stephanos Anderoutsellis-Theotokis and Diomidis Spinellis. "A Survey of Peer-to-Peer Content Distribution Technologies". ACM Computing Surveys, Vol. 36, No. 4, December 2004, pp.335-371.

[14] Adina Crainiceanu, Prakash Linga, Ashwin Machanavajjhala Johannes Gehrke, Jayavel Shanmugasundaram. "An Indexing Framework for Peer to Peer Systems". ACM 1581138598/04/06.

[15] Wolfgang Nejdl, Boris Wolf, Changtao Qu , Stefan Decker , Michael Sintek Ambj orn Naeve, Mikael Nilsson, Matthias Palm´ er , Tore Risch. "EDUTELLA:A P2P Networking Infrastructure Based on RDF". ACM 1581134495/02/0005.