Media Cloud: An Open Cloud Computing Middleware for Content Management

Daniel Díaz-Sánchez, *Member*, IEEE, Florina Almenarez, *Member*, IEEE, Andrés Marín, *Member*, IEEE, Davide Proserpio, *Member*, IEEE, and Patricia Arias Cabarcos, *Member*, IEEE

Abstract — *Cloud computing allows accessing resources across Internet transparently: requiring no expertise in, or control over the underlying infrastructure. There is an increasing interest in sharing media files with family and friends. However, UPnP or DLNA were not designed for media distribution beyond the boundaries of a local network and manage media files through web applications can be tedious. To overcome this problem, we propose Media Cloud, a middleware for Set-top boxes for classifying, searching, and delivering media inside home network and across the cloud that interoperates with UPnP and DLNA*[1].

Index Terms — Cloud computing, content management, content distribution, social networks.

I. INTRODUCTION

Computer paradigms evolved from the mainframe to grid computing, bringing new paradigms that changed our way to use and understand computers. Personal devices and consumer electronics have been influenced by those changes. Cloud computing is a new paradigm that offers scalability, reliability, availability when accessing resources across Internet. Moreover cloud computing is expected to abstract the details of the underlying infrastructure even when they are complex. The term "cloud" is a metaphor for the Internet, the network over which different organizations join to dynamically offer scalable resources [1]. Media management is among the most outstanding aspects of cloud computing, since the cloud makes possible to retain and share large amounts of digital media.

Cloud computing might be a good solution for processing content in distributed environments. Current state-of-the-art devices can produce, store and deliver high quality media that can be finally distributed towards social networks and communities where constituent members might be family or friends. However, there is no infrastructure to keep data under control or even find a concrete media in the home environment or outside it. Media cloud has been designed to cope with this problem letting users constitute a cloud with friends, family or with people with the same interests with the sole objective of managing media transparently even if media is located outside their domains.

Universal Plug and Play (UPnP) [2] and Digital Living Network Alliance (DLNA) alleviate the problems of sharing contents among devices in the home network but they lack a mechanism for searching across multiple repositories in parallel. They require users to organize contents in repositories and to define rules for sharing them. Moreover, they were not designed for managing media outside the home domain.

In conclusion, familiar operations for finding and sharing media with devices over the Internet turn finally in an awful lot of clicking, typing, searching, copying, and pasting. To cope with this problem, this article describes a solution for bringing the cloud computing concept to the home domain. The solution describes a middleware that can be instantiated in Set-top boxes (STB) or home gateways, called Media Cloud, for classifying, searching, and sharing media across the home domain and the cloud. Media Cloud uses a plug-in system to support several content management technologies and it can be extended to future technologies. In fact, already deployed protocols as UPnP and DLNA can be used transparently. The objective is to provide the maximum compatibility with the less effort from user side. For that reason, it aligns to cloud computing concepts so users no longer need expertise in, or control over, the underlying technology.

The remaining of the article is organized as follows. Section II briefly introduces the cloud computing paradigm and its relation to content management. In section III, we describe the objectives of Media Cloud. Section IV introduces the Media Cloud architecture that will be debriefed in section V, that describes the interface to the home network and section VI that describes the interaction with other instances in the cloud. Section VII depicts the features of the security layer of Media Cloud. The implementation details are explained in section VIII together with some performance measures for a deployment on a state-of-the-art STB. Section IX summarizes the goals of Media Cloud.

II. CLOUD COMPUTING

As suggested in plenty of articles across the scientific literature, cloud computing is envisaged to be the next key computing paradigm. This new approach is expected to bring

[1] This work was supported in part by Celtic Netlab project.

Daniel Díaz-Sánchez is with the Telematic Eng. Department, Carlos III Univ., 28911, Leganés, Madrid, SPAIN (e-mail: dds@it.uc3m.es).

Florina Almenarez is with the Telematic Eng. Department, Carlos III Univ., 28911, Leganés, Madrid, SPAIN (e-mail: florina@it.uc3m.es).

Andrés Marín is with the Telematic Eng. Department, Carlos III Univ., 28911, Leganés, Madrid, SPAIN (e-mail: amarin@it.uc3m.es).

Davide Proserpio is with the Telematic Eng. Department, Carlos III Univ., 28911, Leganés, Madrid, SPAIN (e-mail: dproserpio@inv.it.uc3m.es).

Patricia Arias Cabarcos is with the Telematic Engineering Department, Carlos III Univ, 28911, Leganés, Madrid, SPAIN (e-mail: ariasp@it.uc3m.es).

Contributed Paper
Manuscript received 04/12/11
Current version published 06/27/11
Electronic version published 06/27/11.

scalability, availability, and ubiquitous access as never seen before. However, despite this growing interest in Cloud Computing, there are many voices pointing to the lack of an accepted definition for this computing paradigm [3].

The core concept behind Cloud Computing is Software as a Service (SaaS). Cloud Computing, and its complexity, born from squeezing or generalizing the SaaS concept to exhaustion. According to this, if in SaaS an application can be a service, also does the environment over which the application is executed, and even the hardware that executes the entire software. Following this reasoning, Cloud Computing is a resource aggregation of applications, components, frameworks… that can be configured for serving several purposes.

When it comes to the user role, the interaction with Cloud Computing systems might be similar to already existing paradigms. Reference [4] shows the evolution of computing in the last century and how it relates to Cloud Computing. In the very beginning, a mainframe (a central server shared by many people) delivered services to small computers. Then personal computers became more powerful bearing any daily task. The popularization of networks and Internet brought the ability to access local network and Internet applications seamlessly. Finally, grid computing facilitated to share processing power and storage among several machines.

The Cloud Computing concept enables the exploitation of resources across Internet. As pointed in [4], Cloud Computing could be seen as an evolution of grid computing where resources are no longer limited to processing power and storage but anything. However, Cloud Computing delivers those services in a simple way, requiring no expertise or control over the underlying infrastructure. Thus, the interface to the cloud might abstract the underlying complexity leading to a concept similar to the mainframe where the user interacts with a big machine.

In few words, cloud computing allows accessing resources across the Internet transparently, in a simple way and providing high scalability and availability.

A. Cloud Computing and Contents

Several approaches employ Cloud Computing technology for content management. Some of them concentrate on collaboration among a big population since cloud computing provides high scalability. Others deal with media processing, using the cloud for reducing processing time and cost applying economy of scale.

Internet has revolutionized the TV scenario diffusing the frontiers that used to separate TV stakeholders as producers, distributors and consumers. Customers consume user-generated contents, linear TV, or Video on Demand (VoD) anywhere through broadcast, distribution networks, or Internet. Moreover, average users, especially new generations, also known as digital natives [5], have become active broadcasters uploading videos to Internet. This scenario presents several challenges to be solved, especially those regarding content processing, storage and distribution.

Reference [6] signals the plethora of media displays as one of the major challenges of this scenario. Different device resolutions, qualities and form factors, require generating different versions of the same content showing that transcoding might be one of the most critical tasks to be conducted. This entails huge doses of media processing that is computationally expensive. The article sketches out "Split and Merge", a cloud based platform that distributes and parallelizes the video encoding process in order to reduce video-encoding times. Some providers offer cloud-based encoding services to third parties through Internet.

The solutions that lean on Cloud Computing to improve cooperation seek more efficient ways to create, manage, distribute, and archive content. Those solutions could be classified in corporate and open. Corporate solutions aim at providing high scalability, reliability, and simplicity to manage content generated during the business process. Reference [7] describes a prominent approach for content management under the scope of one or several companies. The approach defines content spaces as a way to make cross-organizational content-centered collaboration, seamless manipulation, synchronization of repositories and flexible user management with the aid of the cloud.

On the other side, open solutions could be those targeting systems where constituent members can be end-users. The objective of these open solutions is to help users to create a community for managing contents located inside or outside their local networks. The originality of this approach is to make the user equipment part of the cloud instead of hiring cloud services from third parties. Moreover, this approach is especially meaningful when the cloud deals with sensitive content, as user generated content, since trust in private clouds, as Media Cloud, is feasible to achieve and maintain.

III. MEDIA CLOUD OBJECTIVES

Media Cloud is a middleware for enabling media-centered cooperation among home networks. Media Cloud is the bridge to an open architecture that allows users to join their home equipments to constitute a cloud. Media Cloud abstracts the underlying complexity to provide a new content distribution model that simplifies classifying, searching and accessing user-generated and commercial content within the home networks. Media Cloud pursues fulfilling three goals.

The first goal is content classification. An average user generates contents very quickly and stores them away in several devices. In fact, users' media library stops growing only since they reach their storage capacity until they buy an additional hard drive or computer; or they just upgrade their hard drive equipped STB. Thus, it is usual for a user to expend big time trying to find contents that were previously stored in his/her own devices, organize them into collections, and manipulate them to produce new formats or presentations. Media Cloud alleviates the problem providing an indexing service for searching, a set of functions over common protocols to add or annotate contents and a user interface to manage them (move, copy, delete).

The second goal is to solve the problem of sharing large amounts of media with family and friends. As it has been shown during the web 2.0 advent, the web has dived in the social plane very quickly. Nowadays, applications are commonly linked to social networks and that principle extends to media sharing applications. However, several privacy problems restrain users to upload personal pictures or videos to social networks. For instance, the Security Research Computer Laboratory at the University of Cambridge revealed in their blog entry "The attack of the Zombie Photos" that many social networks fail to delete personal pictures when instructed to do so by the owners. Moreover, in some cases it is possible to obtain unauthorized access to photos.

Even in the case of having those privacy problems solved, social networks are not ready to cope with large amounts of media files or to provide a comfortable user interface for media files manipulation. For instance, consider a group of friends that have just arrived from a half month travel and they all want to exchange the pictures stored in their cameras. An option would be to upload every picture and video to a social network to let others to individually select and download the pictures they like. However, viewing, selecting and downloading individual media file from the social network would be pretty tough and time consuming apart from the privacy concerns it might rise.

As mentioned before, Media Cloud provides an automatic indexing service for searching and accessing contents spread in devices connected to the home network. When several users decide to join their home networks, the contents indexed by Media Cloud become accessible to the other members according to security policies. Media Cloud provides distributed search and transparent access so contents stored in other home networks appears as if they were part of the home network.

Media Cloud eases content sharing, even if the content is protected, respecting licenses. Commercial content is usually protected by Digital Rights Management (DRM) and copy protection systems to prevent unauthorized distribution. There are several technologies, as DVB Content Protection & Copy Management (CPCM) [8], that permits consuming commercial content in different devices whenever they belong to the user and it seems to be the trend for the following years. To define the boundaries of the license, devices can be grouped together in an Authorized Domain. The Authorized Domain limits the content protection boundaries according to the Usage State Information (license) of the content. These technologies manage content from acquisition until final consumption or exportation according to the particular usage rules of that content [9].

When it comes to commercial content, the goal of Media Cloud is to act as discovery service and license proxy. It just finds contents, exchange licenses and provides a tunnel for communicating protected devices. Thus, a constituent member of the cloud can access commercial content retained by other home network, whenever he/she holds the appropriate license. This is an alternative commercial content distribution that respects content licenses while benefits the distribution since user can get contents from providers or members of Media Cloud.

This article intends to present Media Cloud architecture, its distributed search engine and the content adaptation modules. However, Media Cloud functionality is not limited to this description. Though it is not covered in the article, the middleware takes advantage of the processing power for executing user mushups or transcoding content. It also implements a distributed cache to increase performance.

IV. MEDIA CLOUD ARCHITECTURE

Media Cloud middleware provides services to the devices located inside the home network and to other Media Cloud instances located outside whenever they belong to the same cloud. For that reason, Media Cloud is located between the home network and Internet. An appropriate place to instantiate Media Cloud is an STB with access to the home network and to the Internet. Thus, it can communicate with devices located in the home environment and provide search services, content delivery, and filtering to friends and family outside home domain. Fig. 1 sketches out the architecture of Media Cloud.

Two different modules compose Media Cloud: the Media Indexer and the Foreign Content Aggregator. Aside those modules there is a security layer which enforces security policies and filter contents.

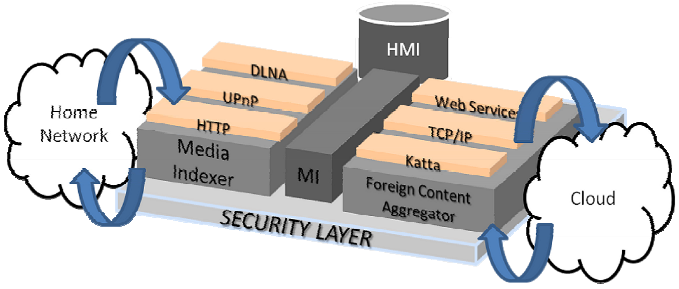

Fig. 1. Media Cloud architecture. The figure shows two functional modules, the Media Indexer and the Foreign Content Aggregator. The security layer interacts with both modules.

The Media Indexer manages communications with devices inside the home domain. It discovers devices, obtains metadata from the media files offered by those devices, builds search indexes, and adapts incoming and outgoing streams appropriately for every device. The most important task of the Media Indexer is the creation of the search index (content cataloging). The index contains metadata, information about the devices, access control information, and any other optional attribute.

The Foreign Content Aggregator attains the goal of making the home network part of a cloud. It extends Katta, an open source distributed application that enables search operations in a similar way as Hadoop Map Reduce [10] does. The Foreign Content Aggregator straighten out the problem of searching across multiple repositories in parallel (one per constituent home network) and also handles incoming or outgoing content streams.

Due to the inherent private nature of user-generated contents, it becomes mandatory the use of a fine-grained access control system based on policies and strong authentication. Moreover, despite Media Cloud connects home networks transparently, there might be several users per household accessing contents thought Media Cloud, so it is necessary to authenticate and authorize users instead of home networks. Media Cloud tackles security threats using digital identity that allows to not only authenticate and authorize, but also to personalize services. The security layer performs delegation for a better performance.

Concerning the entities within a home network, Media Cloud interacts with the underlying hardware and other networked appliances acting as a broker to deliver multimedia inside or outside the home domain. Multimedia files can be stored in any device within the home domain. These devices must be able to communicate with other devices by means of DLNA, UPnP or any other media sharing protocol. To make multimedia files stored by a device visible to the cloud using Media Cloud, devices can either implement a service for metadata extraction, instantiate a Media Cloud device agent (a small service for metadata extraction), or let Media Cloud fetch the beginning of multimedia files since metadata is usually stored in that region.

The Media Cloud middleware (instantiated in an STB) collects metadata, provides searching services, and acts as a proxy adapting requests to protocols supported by media endpoints. Thus, it just disguises the complexity of the Internet to the home domain devices and, obviously, to their users.

V. MEDIA INDEXER

The Media Indexer discovers devices located in the home domain and interrogates them to gather information about the contents they retain. It uses content information to produce an index that will be used by the Foreign Content Aggregator to facilitate search operations. The Media Indexer has two functional blocks: the Content Indexer and the Home Domain Manager.

The Content Indexer collects information about the content as media type, creation date, metadata, and user's annotations. It also registers access information as the hardware identifier or the network address of the device holding that content, protocols that can be used to access the content and required license (in the case of commercial content). With that information, it builds an index and stores it in the Home Media Indexes (HMI) database.

When the Content Indexer processes a media file, as a photo or video, it needs cooperation from the device, for instance, to provide information about the media file. This procedure requires the device to implement a metadata extraction service accessible by, for instance, DLNA or UPnP. Even though that is the preferred way, the Content Indexer can fetch the media file from the device or part of it, since the beginning usually contains the metadata, to process it.

As the reader may infer, supporting the plethora of devices and protocols present in the market and coping with upcoming technologies is a tough task. For that reason, a separate module called Home Domain Manager (HDM) handles interaction with devices. The HDM relies on a plug-in system for supporting different devices and protocols and new plug-ins can be developed to support new hardware, protocol, or technology.

A. Content Annotations

Nowadays multimedia devices allow users to annotate content including, for instance, a text annotation about the event covered by a given media. State-of-the-art cameras configuration permits to provide some text that would be embedded in every media file produced by the camera. Moreover, digital cameras and scanners include some metadata by themselves.

The Exchangeable Image Format (Exif), which was created by the Japan Electronic Industry Association, uses the JPEG and TIFF formats to include metadata tags. These tags might contain date and time information, technical information as camera model, aperture, ISO speed, thumbnails for previewing the picture, description and copyright, or even geolocation information provided by a GPS receiver (geotagging).

The International Press Telecommunications Council (IPTC) makes recommendations for including structured metadata within electronic media as images, audio and video. The IPTC metadata covers author, caption, comments and many others. The JPEG File Interchange Formant (JFIF) defines metadata compatible with Adobe Photoshop JPEG Information Resource Block extension and IPTC but incompatible with Exif. However, digital cameras produce pictures with both annotations.

There are also other annotations for multimedia including audio and video. IPTC recommendations target any media including audio and video. However, other annotations as ID3 are more popular. ID3 is a de-facto standard for cataloging multimedia using metadata in audiovisual containers. ID3 version 2.0 defines more than 30 standard Unicode encoded tags of unlimited length that includes lyrics, pictures of the cover, author information, artist details, album title…

These tagging systems have been inherited by modern multimedia containers as Matroska, Ogg or AVI, which might contain fully annotated video with several audio tracks, subtitles, images of the cover and many others. Metadata is very useful for cataloging multimedia and to enable text based search. However, annotations in Media Cloud are not limited to the aforementioned systems.

Media Cloud allows complementing that information with social networks information and related Internet content. The middleware relies on a "Social Enabler" to fetch feeds related to a media file from social networks. In [11] we described a Social Enabler that uses content metadata to find feeds from friends in social networks. In such a way, Media Cloud coalesces friends' comments about the media into a document and updates it regularly.

B. Content Indexer

The Content Indexer (CI) is in charge of building an index for contents using content metadata, social network feeds and Internet related content (see Fig. 2). This module uses Lucene, an open source scalable high-performance indexer that enables searching over the index using ranked or fielded searches. It is possible to use many different queries as phrase, wildcard, proximity, or range queries.

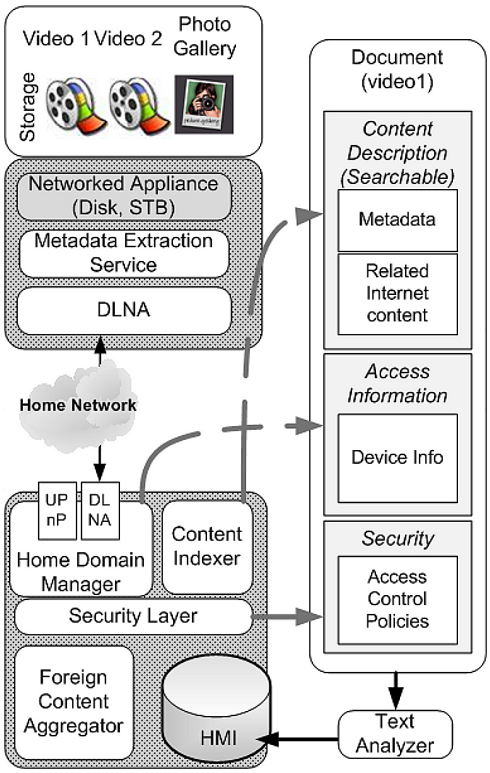

Fig. 2. Content Indexing. This figure sketches out the structure of an index entry (document) and the contributions of the different modules.

A Lucene index is a directory. Every index entry corresponds to a document inside the directory. The CI creates a document for every media file analyzed and adds it to the index. A document is a collection of field-value pairs. The number and nature of fields depends on the media file since the information extracted from metadata, Internet or social networks might be different.

Fields in Lucene can be stored, tokenized, indexed, and vectored. Stored fields contain the value as it was provided to Lucene. In tokenized fields, the value is analyzed and tokens emitted are indexed. The value in indexed fields is made searchable. Finally, vectored fields contain the term frequency per document.

The fields of the document generated by CI for every media file can be classified in three categories. The content description category contains a field per metadata entry in the file. Since the field name must be unique within a document, the CI uses a namespace name as a prefix for every metadata entry, for instance, "id3.title" or "ipct.author". The fields belonging to this category are indexed and stored in the document, so it is possible to search across them.

The access information category contains fields required by the Home Domain Manager to retrieve the content from the device. The fields of this category contain hardware information, network addresses, ports, and protocol details to interact with the device retaining the content within the home network. These fields are stored in the document but not indexed.

The security category contains fields with access control information and optionally license information (in case of commercial content). The access control information limits the usage of the content, for instance, who can access it, if it is shared in the cloud or accessible only in the home network… The preferred policy language is XACML [12] but Media Cloud can be easily extended to other policy languages.

C. Home Domain Manager

The Home Domain Manager (HDM) deals with the different networked devices present in the home network. It relies on a plug-in system for supporting different devices and protocols, and it can be extended to support upcoming technologies. When contents are requested from the home network or from the cloud the HDM retrieves the access information fields from the index and instantiates the appropriate plug-in (see Fig. 3).

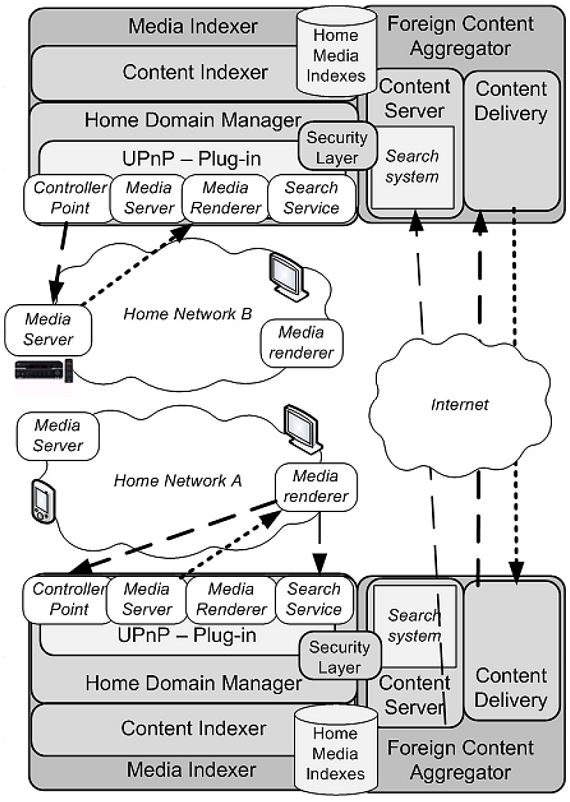

Fig. 3. The Home Domain Manager abstracts the complexity of the cloud to the devices in the home network by using different plug-ins. The figure shows a scenario where two UPnP devices located in different home networks interact. The HDM on home network A acts as a media server for the device and the HDM on home network B as a media renderer. The content stream is delivered through Internet.

This plug-in system deals with the plethora of technologies available for content distribution within a home domain. UPnP and DLNA are supported by the HDM, but it can be extended to other protocols. DLNA and UPnP deal with networked consumer electronics permitting user-generated contents to be shared among household devices. These specifications define three functional components: Media Server (MS), Media Renderer (MR), and Control Point (CP). A device can implement several functional components (media players combine CP and MR). Control Points discover and control other devices on the network and coordinate operations among devices that yield to the desired result. Devices in DLNA expose services that provide actions. Services can be controlled via state variables or events.

UPnP AV facilitates the discovery and configuration but it does not define how contents are transferred. DLNA goes beyond UPnP defining mandatory Media Formats and Media Transport protocols as HTTP or Real-time Transport Protocol (RTP). However, distributed search operations in UPnP and DLNA are not straightforward. UPnP behaves in a Peer-to-Peer (P2P) fashion, for instance, a CP controls a MS to render contents in a MR, so UPnP does not allow to search in parallel in several repositories. The Content Indexer and the Foreign Content Aggregator handle the searching operations.

The HDM acts as a broker. The module gets the content stream from the source device using the appropriate plug-in. When Media Cloud handles communications within the home domain, the content stream is redirected to the destination device. If source and destination devices use a different protocol, the HDM instantiates two plug-ins, one to receive contents from the source device and another to send the content to the destination device. The content can be transcoded if necessary. When Media Cloud handles communications with the cloud, the content stream is redirected to the Foreign Content Aggregator, which eventually will send it through Internet to the destination.

VI. Foreign Content Aggregator

The Foreign Content Aggregator handles cloud communications. It makes content stored in devices at the home network available to other Media Cloud instances through Internet. The module is composed by a Content Server and a Content Delivery module.

The Content Server facilitates foreign clients to search within the HMI database. The Content Delivery module sends content to other Media Cloud instances located outside the home domain.

The authentication is handled by the Security Layer that issues a security token after a foreign client is successfully authenticated and authorized. The Access Control System of the security layer uses the security token to filter HMI database contents preventing unauthorized access and respecting privacy.

A. Content Server

UPnP and DLNA, as many other protocols used to share contents within a home domain were designed to operate in local networks. For that reason, users rely frequently on third party services, typically web-based, to share contents beyond the boundaries of a home domain. These kinds of workarounds are orthogonal to cloud computing concepts since contents are not accessed transparently.

Fortunately, many initiatives allow clients to perform distributed search operations by connecting to all nodes and merging results into a unified result list. Those initiatives employ "map and reduce" functions [13] commonly used in functional programming.

The Foreign Content Aggregator is based on Katta, a distributed application that runs on commodity hardware. Katta requires a master server to manage the rest of the nodes of Media Cloud. Nodes are participants of the Media Cloud that serve index "shards".

The index shards are generated from the Lucene indexes stored in the HMI database. A member of the cloud can search within an index since Katta connects to all the members of the Media Cloud and merges results into a unified result list. Thus, devices within the home network can search content using Media Cloud in several repositories in parallel as if they were part of their home network.

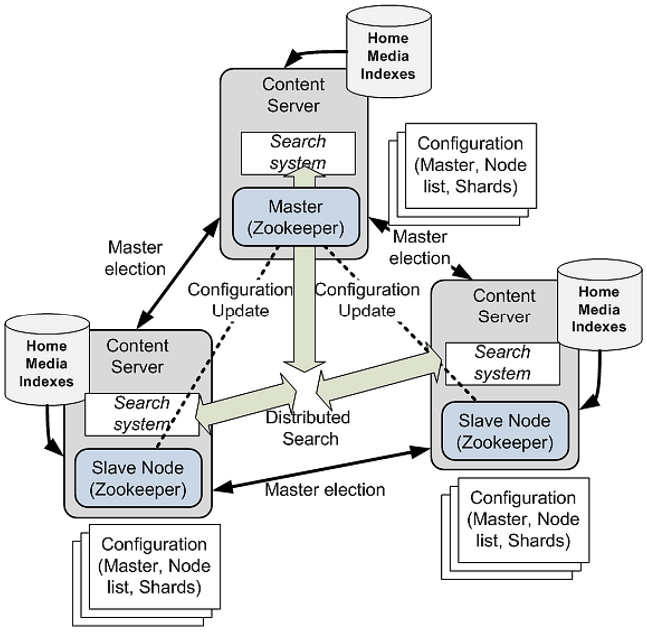

Fig. 4. Content Server and its relation with other Media Cloud instances. The master election is performed during the start up of the system or whenever the master fails. Zookeeper maintains the configuration updated that includes node list, current master and shards.

The Media Cloud middleware can act as a Katta master or a node. One Media Cloud instance should act as a master. The master is drawn during the start up. If the master fails, the nodes start the draw again to continue the operation.

Katta uses Zookeeper, a centralized service (where the central node is the master) for maintaining configuration

information, naming, providing distributed synchronization, and providing group services. Zookeeper keeps track of the live nodes and updates the node list in every Media Cloud instance when a node fails or a new node joins the cloud.

When a search operation is performed, Katta gets the document frequencies, i.e. number of times the word(s) of the query is(are) contained in a document, for a query individually from all the nodes. That gives the document frequencies per node, but not a global score (or frequency). Then, it passes the value (document frequency) and the search query to all nodes so they can adjust their scoring in order to derive the document frequency, or scoring, with a global scope. In this way, it is possible to obtain a distributed scoring system to find the contents that better match the search query.

B. *Content Delivery Module*

The Content Delivery Module (CDM) handles communications with foreign devices. This module delivers content outside the home network by means of streaming, http or any proprietary protocol using a general-purpose secure tunnel.

The module selects the most appropriate protocol to send contents across Internet. The device retaining the selected content streams it to the Home Domain Manager. The CDM at the source home network provides an appropriate transport to the content over a secure tunnel. The CDM at the destination receives the content and redirects it to its Home Domain Manager. Finally, the Home Domain Manager at destination Media Cloud instance, would select the appropriate plug-in to deliver the content to the device that requested it.

The CDM uses a plug-in system that can be extended to support new protocols. By default, the CDM supports RTP and Real Time Streaming Protocol (RTSP) protocols for streaming. It also supports HTTP and HTTP over secure channel (HTTPS) protocols for transmitting content that cannot be streamed as images or documents. If the protocol for communicating two devices in different home networks through the cloud is proprietary, the CDM provides a general-purpose secure tunnel that acts as secure pipe.

Devices located at the home network can access transparently to contents stored in the cloud as if they were part of the home network. The Home Domain Manager and the Content Delivery Module perform content streaming and adaptation. The Content Delivery Module sends or receives contents from other Media Cloud instances and the Home Domain Manager adapts the streams to fit devices capabilities. Media Cloud abstracts the underlying complexity so the devices interact as if they were located in the same home network.

VII. SECURITY LAYER

The security layer is among the most important pieces of Media Cloud. There are several concerns about security in cloud computing especially when user-generated content can be delivered, stored and processed in nodes outside the administrative boundaries of the user domain. In [14] and [15] could computing security is analyzed unveiling that the most important problem is trust. Those articles reason about public Cloud Computing systems that offer services to end users or other companies. In general, trust is a trade off against many benefits as scalability, performance and ease of management. But when it comes to user generated content, which can comprise sensitive content, trust is absolutely necessary.

The problem of trust in Media Cloud is alleviated since the cloud could be considered private. Media Cloud is managed by a community where constituent members might be family, friends or anyone explicitly invited to be part of it.

The security in Media Cloud is based on digital identity. A Media Cloud instance provides services to devices located in the home network. Those devices can be operated by any family member. Thus, the authentication, authorization and policy enforcement should be managed using user-centric digital identity technology.

Modern user-centric digital identity can be defined as "what I say about me, and what others say about me" [16], since users coalesce attributes from different places ad-hoc for each interaction, keeping entire control over their data. Some user-centric digital identity paradigms, as Information Cards [17], permit to build personalized cards, metaphors of real ID cards. The idea is not only to authenticate and authorize Media Cloud users but also to personalize the service, for instance, filtering contents or enforcing parental control.

Media Cloud relies on Information Cards to perform authentication and authorization. When a user starts using Media Cloud, he requests his Media Cloud STB to generate an Information Card and to send it to the members of the cloud or to those offering the services requested by the user. Upon reception, Media Cloud nodes perform authentication and authorization on the Information Card. If the authentication was successfully and the requested services fit on the authorization policy, each node issues an OAuth [18] token that will be used for further interaction. OAuth allows Media Cloud nodes to delegate resources to other node or device without requiring to hand out credentials at the beginning of every interaction. In such a way, every Media Cloud node delegates part of their functionality to the token holder.

The Security Layer uses the security token to filter HMI database contents limiting the search results and the access to the contents. Thus, every user has a different view of the cloud that depends on the permissions granted by other nodes to access their content.

VIII. IMPLEMENTATION

We have developed Media Cloud including an UPnP service for metadata exchange that can be easily instantiated in small devices. We have chosen an open source UPnP library for developing the UPnP/DLNA plug-ins of the Home Domain Manager and for the metadata exchange service.

The development process of Media Cloud comprises three stages. In the first stage, we developed a proof of concept using commodity hardware as Personal Computers (PCs). In this

stage, Media Cloud was successfully instantiated and tested in several J2EE containers. Concerning the hardware, we tested Media Cloud in a small form factor PC with 1Gb of RAM.

The Home Media Indexes database was developed using the Apache Lucene open source project. To make the index searchable across the cloud, we implemented a custom Content Server with a distributed search library from the Katta project. Our custom Content Server handles the master node election during the initialization and upon master failure.

The security layer in this stage used an open source XACML implementation for policy enforcement and an open source security framework for user management, authentication, authorization, and policy enforcement. The information cards authentication was implemented as a custom authentication module for the security framework.

In the second stage, we faced the challenge of making a stable implementation with a low memory footprint to be instantiated in a state-of-the-art STB with a Reduced Instruction Set Computing (RISC) processor, 256Mb of RAM, and a Gigabit Ethernet network interface running Linux operating system.

Media Cloud offers a search service and acts as a proxy for delivering content inside or outside the home network. The content delivery service is limited by the bandwidth available to the home network and the processing power if transcoding is requested. The Media Cloud monitors the bandwidth preventing greedy foreign nodes to hoard the uplink and uses a priority schema for queuing requests. However, this operation consumes few resources since it requires only encapsulating packets from inside the home domain appropriately for delivery over the Internet or the other way around. Concerning transcoding, Media Cloud plug-ins rely on hardware for those purposes limiting the CPU usage when there is no special hardware available for performing transcoding.

Content delivery and transcoding are limited as mentioned before, however, searching operations over a Media Cloud node might be very frequent. The number of members of the cloud can be very large so the Media Cloud STB might receive multiple simultaneous searching operations. The main objective of the Media Cloud implementation for STBs was to limit the memory and processor consumption of those search operations. We implemented two different modes of operation for the searching services to serve requests either from inside and outside the home network: relaxed and loaded.

The major constraint of the searching service is RAM since processing a request requires allocating memory for the request and for the associated security policy, whereas internal search operation over the index uses resources already allocated. Searching over the index takes usually the same time so the difference between modes is the RAM they allocate for this purpose. In the relaxed mode, we dedicate 8 Mb of RAM to the searching services limiting the CPU consumption to 20%. The loaded mode consumes 16 Mb of RAM and a 25% of the CPU.

We tested our implementation with a STB for both modes of operation. The scenario was a cloud with 50 participants making one request at the same time to the Media Cloud node instantiated in the STB. We repeated the process 50 times. The Media Cloud STB used an index derived from 10 thousand media files. The memory consumption and the test time are shown in Fig. 5 for the relaxed mode and in Fig. 6 for the loaded mode.

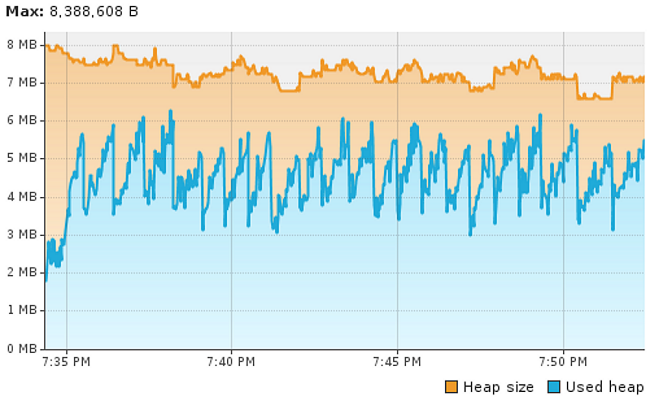

Fig. 5. Test results for relaxed mode using 8Mb of RAM.

The test for the relaxed mode lasted 1102 seconds. The Content Server used 547 seconds of that time to process the 2500 requests. The rest of the CPU time was used to parse the request, check the security policy, generate the response, and send it. The average used heap was about 5.6 Mb.

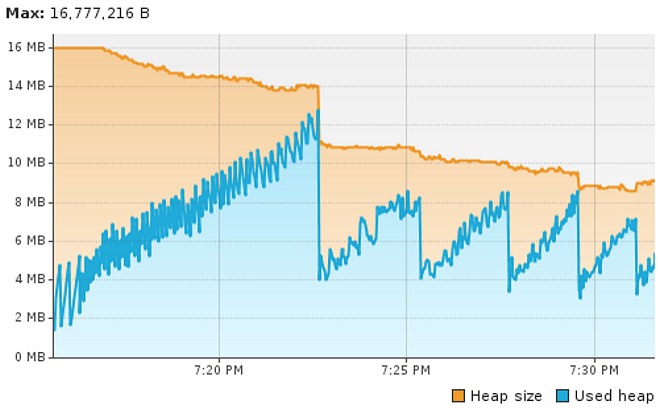

Fig. 6. Test results for loaded mode using 16Mb of RAM.

The test for the loaded mode lasted 950 seconds. The Content Server used 541 seconds of that time to process the 2500 requests, which is reasonably similar to the time the Content Server used in the relaxed mode tests, showing that the search operation time is, in practice, constant. Besides the average used heap was also around 5.6 Mb, the bigger heap allows Media Cloud to accommodate more requests without needing to free part of the heap making this mode a 15% faster.

The security layer has been developed using a minimal version of the first stage security framework in order to reduce the memory footprint. Regarding the performance, user

authentication/authorization and Information Card generation tasks consume few resources and are executed few times thus the impact on the performance is very small.

We are currently on the third stage of the implementation process that comprises the implementation of plug-ins for other protocols than DLNA or UPnP and testing hardware for handling commercial content.

IX. CONCLUSIONS

Media Cloud provides an easy to manage, cost-effective solution for bringing cloud computing paradigm to content sharing among federated home networks.

The solution is easy to manage since it supports different devices by performing content adaptation. Media Cloud considers transparency as a main goal: it allows devices from different home networks to communicate as if they were in the same local network. It uses well known protocols as DLNA and UPnP for interfacing the home network whereas uses HTTP and RTP over a secure channel for communications across Internet. Moreover, the solution is open since new protocols can be supported using the plug-in system.

The cost effectiveness is achieved by sharing resources that could be underused in other cases. Media Cloud encourages cooperation among home networks facilitating media classification, management and sharing. Distributed search and content delivery over the cloud are among the most important features of Media Cloud.

Unlike other cloud computing solutions, Media Cloud, due to its private character, mitigates privacy problems. It relies on digital identity to perform per user authentication and personalization, and uses OAuth tokens for filtering search results and perform access control.

REFERENCES

[1] A. Weiss, "Computing in the clouds," *netWorker*, vol. 11, no. 4, pp. 16-25, Dec., 2007.

[2] A. Presser et al., "UPnP Device Architecture Version 1.1," UPnP Forum Tech. Rep. V1.1, October 2008

[3] H. Erdogmus, "Cloud Computing: does nirvana hide behind the nebula?," *IEEE Software*, Vol. 26, no. 2, pp. 4-5, Mar., 2009.

[4] J. Voas and J. Zhang, "Cloud Computing: new wine or just a new bottle?," *IT Prof.*, Vol. 11, no. 2, pp. 15-17, Mar., 2009.

[5] U. Gasser and J. Palfrey, *Born Digital: Understanding the First Generation of Digital Natives*, Basic Books, NY, September 2008.

[6] K. Breitman, M. Endler, R. Pereira, and M. Azambuja, "When TV Dies, Will It Go to the Cloud?," *IEEE Computer,* Vol. 43, no. 4, April 2010.

[7] J.S. Erickson, S. Spence, M. Rhodes, D. Banks, J. Rutherford, E. Simpson, G. Belrose, R. Perry, "Content-Centered collaboration spaces in the cloud," *IEEE Internet Computing*, Vol. 13, no. 5, pp. 34-42, Sep., 2009.

[8] European Telecommunications Standards Institute, "Content Protection and Copy Management Specification; Part 2: CPCM Reference Model", ETSI, Sophia Antipolis, France, Tech. Rep. TS 102 825-2 V1.1.1, July 2008.

[9] European Telecommunications Standards Institute, " Content Protection and Copy Management Specification; Part 3: CPCM Usage State Information", ETSI, Sophia Antipolis, France, Tech. Rep. TS 102 825-3 V1.1.1, July 2008

[10] M. Bhandarkar, "MapReduce programming with apache Hadoop", in *Proc. of IEEE International Symposium on Parallel & Distributed Processing*, May 2010.

[11] D. Diaz-Sanchez, A. Marin, F. Almenarez, A. Cortes, "Social applications in the home network *IEEE Trans. Consumer Electron.*, vol. 56, no. 1, pp. 220-229, Feb. 2010.

[12] OASIS eXtensible Access Control Markup Language Technical Comitee, "XACML 2.0 Core: eXtensible Access Control Markup Language (XACML)", OASIS, MA, Tech. Rep. access_control-xacml-2.0-core-spec-os, 2005.

[13] J. Dean and S. Ghemawat, "Mapreduce: Simplified data processing on large clusters," in *Proc. of Symposium on Operating Systems Design and Implementation*, pp. 137–150, 2004.

[14] A. Ghosh, and I. Arce, "In Cloud Computing We Trust - But Should We?," *IEEE Security and Privacy*, Vol.8, no. 6, Dec., 2010.

[15] L.M. Kaufman, "Data Security in the World of Cloud Computing," *IEEE Security and Privacy*, Vol. 7, no. 4, Aug., 2010.

[16] Dick Hardt, "Keynote Talk - Identity 2.0", in *Proc. of O'Reilly Open Source Convention*, 2005.

[17] Kim Cameron et al., " Proposal for a Common Identity Framework: A User-Centric Identity Metasystem," Information Cards Foundation, MA, Tech. Rep., Oct., 2005.

[18] E. Hammer-Lahav, "The OAuth 1.0 Protocol", Internet Engineering Task Force (IETF), RFC 5849, Apr. 2010.

BIOGRAPHIES

Diaz-Sanchez, Daniel (M'07) received a Telecom. Eng. degree from Univ. Carlos III de Madrid in 2002. He graduated as Master Telematic Engineering (2004) and obtained his PhD (2008) from Univ. Carlos III of Madrid. He works as researcher and teacher at Universidad Carlos III. His research topic is distributed authentication, authorization and content protection.

Andrés Marín López (M'07) received a Telecom. Eng. degree and PhD from the Technical Univ. of Madrid in 1992 and 1996 respectively. He lectures in Computer Networks and Ubiquitous Computing in the Univ. Carlos III de Madrid, where he has a position as associate professor since 1998. His research interests include ubiquitous computing: limited devices, trust and security services, and security in NGN.

Florina Almenárez Mendoza (M'07) received the Computer Engineer degree from the University Autónoma of Bucaramanga (Columbia) in 1999, and her Ph.D. degree from the University Carlos III of Madrid (Spain) in 2006. She currently works as an associate professor and researcher in the University Carlos III of Madrid. Her research interests include distributed trust management models for dynamic environments, security architectures in pervasive devices, and security for ad hoc networks.

Proserpio, Davide received a Telecom. Eng. degree from Technical University of Milan in March 2008 and a MSc degree in 2010 from the University Carlos III de Madrid. Currently he is a PhD candidate in the Telematic Engineering department of the Carlos III University. His research topics include security in NGN, Advanced Authentication, and Digital Identity Management.

Arias Cabarcos, Patricia received her Telecom. Eng. degree from Univ. Carlos III of Madrid in 2008 and she obtained the MSc degree in Telematics in 2009. Currently, she is pursuing a PhD at the Department of Telematics Engineering in the Univ. Carlos III of Madrid, working within the Pervasive Computing research group. Her research focuses on the problem of identity management in open and dynamic environments, with special attention to risk analysis and the underlying trust models