Tracking Semantic Relationships for Effective Data Management in Home Networks

Ashok Anand, Aaron Gember, Aditya Akella
University of Wisconsin, Madison
Madison, WI, USA
{ashok,agember,akella}@cs.wisc.edu

Vyas Sekar
Carnegie Mellon University
Pittsburgh, PA, USA
vyass@cs.cmu.edu

ABSTRACT

The amount of data that home users generate, store, and peruse has grown significantly in the past few years. Increasingly, organizing this huge amount of data—in order to make it easy to browse, query and access—is becoming challenging. Many recent proposals have emphasized the importance of data management in home networks and proposed mechanisms for managing replicas across devices to increase availability. Essentially, they capture the relationship "is copy of" between files across devices. However, files can be semantically related. Users are often interested in finding data that has such semantic relationships; tracking these relationships helps users to effectively search based on content or human-understandable context, organize data and manage the limited storage while ensuring availability of information. However, inferring semantic relationships just based on user-defined tags and file names can be challenging, since users may not follow any standard or unique naming conventions. We argue that such semantic relationships should be derived on the basis of content itself, and propose to leverage recent developments in multimedia processing literature, with minimal user involvement. The decentralized, heterogeneous and dynamic operational environment of home networks present interesting systems and network challenges. In this paper, we have highlighted several candidate designs and system- optimizations that can help build an effective semantic-aware data management for home networks. As ongoing work, we are working on a prototype implementation of a decentralized data management system.

Categories and Subject Descriptors

C.2 [**Computer Communication Networks**]: Miscellaneous; H.4 [**Information Systems Applications**]: Miscellaneous

General Terms

Management, Design

HomeNets 2010, September 3, 2010, New Delhi, India.

1. INTRODUCTION

The amount of data home users generate, store, and peruse has grown significantly in the past few years. Home users are storing huge volumes of multimedia content – photos, music, and videos – in a growing number of devices including desktops, laptops, mobile phones, "media centers" [6], and cameras. Industry estimates suggest, by this year, users in a typical networked home will be required to deal with terabytes of data, in hundreds and thousands of files, stored and used across ten or more devices [2]. Increasingly, organizing this flood of data—to make it easy to browse, query, and access—is becoming challenging. Furthermore, home networks present unique challenges in that home users possess neither the technical expertise nor the management resources to instrument data management solutions typical of enterprise networks [22].

Many recent proposals (e.g., Perspective [23] and Cimbiosis [19]), have emphasized the importance of data management in home networks and proposed mechanisms for managing replicas across devices to increase availability. These approaches capture the relationship "*is copy of*" between files across devices. However, files can also be *semantically* related. For example, two videos can have the same content but different resolutions or encoding rates. As another example, a user may have multiple images of the same event or person. Users are often interested in finding content that has such semantic relationships; thus directly supporting the ability to track such relationships helps meet user requirements more directly. Tracking these relationships could also help carefully manage the limited storage across the devices while ensuring high availability of information.

Inferring semantic relationships can be challenging. Home users acquire media content from many sources (e.g., friends, other family members, Internet) with diverse presentation formats. There will likely be semantic relationships across such media files that are unknown to users. These relationships cannot be discovered with user-defined tags and file names, since users may not follow a standard or unique naming scheme. For example, a user would not know whether a downloaded song on her mobile phone already exists on her laptop in a better format, if the file names differ. If such relationships are known, she can delete the song to free up space in the mobile phone and need not transfer it to the laptop. The user could also obtain the better format of the song from her laptop, instead of downloading the song again from the wide area network. Similarly, if tags associated with photos are not standardized, it becomes difficult to determine whether a photo in Bob's mobile phone and a

photo in Alice's mobile phone were taken during the same event.

Based on these observations, we argue each file should be associated with certain *semantic features* that are generated based on *the content itself*. That is, rather than rely on unreliable techniques like file names and tags, the features capture an intrinsic *perceptual* property of the data. For example, these semantic features could be used to generate content fingerprints for identifying songs and videos; recognizing objects, people, places, and events in images; etc. Fortunately, recent developments in the multimedia processing literature [13, 15] enable us to automatically infer such semantic relationships with minimal user effort and involvement.

While the building blocks for inferring semantic relationships are in place, realizing a practical semantic-aware data management system for home networks presents interesting networking and system-level challenges:

- **Decentralized operation:** User surveys indicate that home users users are not willing to invest time or money in building and managing a dedicated data management server [22]. Thus, we cannot assume that there is a centralized file-server.
- **Heterogeneous devices:** Home devices possess different hardware resource constraints and software capabilities. For example, mobile phones are energy and processing constrained. Tasks like feature generation and responding to search queries should take into account device resource constraints.
- **Dynamic environment:** Files may not always be synchronized from mobile/handheld devices to more stable laptops/desktops. Thus, it is necessary to allow user queries to be satisfied directly by all types of devices.
- **Intermittent connectivity:** Home devices may not be connected at all times and connectivity quality may vary. Search and delivery mechanisms should take into account connectivity and network properties.

In the rest of this paper we discuss these issues in greater detail and present various strawman ideas for addressing these challenges.

2. BENEFITS OF SEMANTIC-AWARE DATA MANAGEMENT

In this section, we discuss how semantic relationships can simplify and aid content management in home networks.

- **Rich search:** Semantic relationships help users to effectively search based on content or other human-understandable context—e.g., finding images that are similar to a specific vacation picture or finding pictures of a specific person. The specific capabilities depend on the types of semantic features supported.
- **Data organization:** Semantic relationships can help users discover related files spread across devices in the household. The user can then group and label data appropriately—e.g., moving content corresponding to a vacation into a single folder on a desktop or creating a single "view" for all related files across devices [23].

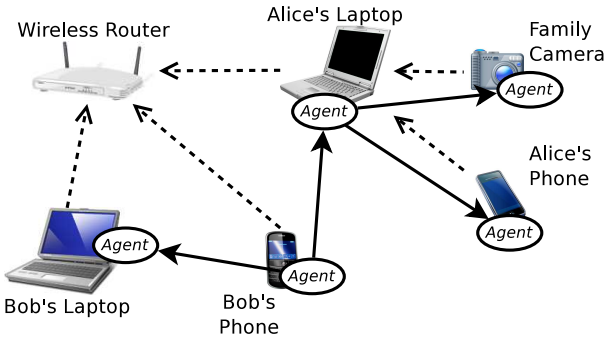

Figure 1: Decentralized home network architecture with each device running a lightweight data management agent. The solid lines represent a query request from Bob's phone.

- **Flexible storage management:** If some files in a device have already been copied to another device, users can simply delete the duplicate copies. Knowledge of semantic relationships adds another dimension to this flexibility. If some of the files have alternate versions available elsewhere, a user may still go ahead and delete the file. In the same vein, users may choose to gracefully degrade old content by encoding them into lower resolution versions to save space. Knowledge of whether two files represent the same logical event can also be useful, as the user may be interested in keeping just a few files corresponding to that event.
- **Resource-aware data delivery:** Semantic relationships can also aid in delivering data to specific devices in a resource-aware fashion. For example, users streaming a video from a desktop to their mobile phone or handheld consoles can switch to lower-resolution versions of the video – if available from other devices – when running low on battery.

In the next section, we describe our specific design and the semantic features we hope to leverage.

3. OUR DESIGN PROPOSAL

We envision a decentralized architecture with each home device running a lightweight data management agent (Figure 1). Each device is connected to either a central network access point or another device, with flux in the connectivity and availability of devices. Our high-level goal is to minimize user effort and involvement. We abstract away low-level system issues and optimizations from the user as much as possible. The agent provides a human-understandable interface to allow users to express their data management requirements and policies. The agent is also responsible for three aspects of data management (Figure 2):

1. Relevant semantic feature extraction
2. Efficient search and query capabilities
3. Effective storage management

We discuss each of these components next.

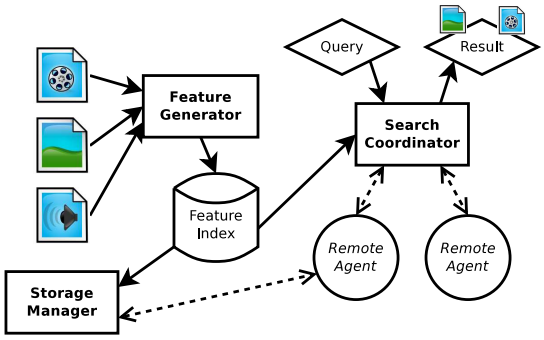

Figure 2: Agent components and their interaction. The dashed lines represent communication with agents on other devices.

3.1 Extracting Semantic Features

Types of features.

There are different aspects of multimedia content upon which users might want to search/query. The specific features would depend on the types of functionality users desire. We discuss a few features that might be useful for many common applications. We use these as representative examples to demonstrate the feasibility of extracting semantic information relevant to different contexts and to highlight the systems challenges that may arise in practice.

1. **Perceptual fingerprints** of the content that are invariant across different presentation formats. For images, these include gradient based techniques [16, 14], and color histograms. For audio content, these techniques are based on signal processing algorithms that mimic human hearing (e.g., [3, 5]). For video content, typical techniques extend the image fingerprints and additionally capture temporal variations [12, 21].

   Most of these approaches, especially in the context of video files, have been proposed for detecting copyright violations. These applications require fingerprinting of all or most of a video file. In contrast, our requirements are less stringent as we are only interested in identifying files that are likely to be related to each other by way of having similar content. This less stringent set of constraints allows us, for example, to apply the aforementioned algorithms to short snippets of videos, or to sampled frames in a video. Of course, we can employ full-video fingerprinting for greater accuracy, but at a higher cost. Another consequence of our less stringent requirements is that we can employ more coarse-grained fingerprinting algorithms that are fast but not accurate. Traditionally, such algorithms (e.g., color histograms) have been shunned because they are susceptible to attacks in an adversarial setting. In our context, since most data is generated and controlled by a collection of home users, the coarse-grained algorithms can serve as practical alternatives.

2. **Contextual information** that qualify key properties of the content. For example, identifying activities, objects, people, or places in images; timestamps; and artists/genres for audio and video content [4].

Computation overhead.

Ideally, we want to use the feature generation algorithm which will give the best accuracy. However, such algorithms tend to be expensive, especially for resource-constrained platforms. For example, computing the color histogram for images takes 300ms while more expensive gradient features like SIFT [14] take close to 32 seconds on a Nokia N810 phone [27]. To address this challenge, we use a combination of three solutions:

- *Device-specific customization:* The key idea is to use different features depending on the specific capabilities of the devices. Since our goal is not to find exact matches, we can trade-off some loss in accuracy and use more coarse-grained features. For example, we can annotate different algorithms using historical profiles of their resource requirements and quality to select the highest fidelity algorithm the device can currently handle [7].

- *Multi-stage computation:* We can extend the device-specific approach to allow a triggered feature generation process. That is, we use lightweight feature generation algorithms by default and generate expensive features only on-demand.

- *Offloading:* For mobile devices, feature generation can be offloaded to nearby laptops or desktops. Here, we can additionally leverage information regarding file replicas to assign offloading responsibilities that minimize communication overhead [23].

Bootstrapping and handling legacy devices.

Some legacy devices may not have the computational resources or may be incompatible in supporting functions for generating semantic features. In this case, we envision some stable and compatible device (e.g., a laptop or desktop) that acts as a proxy for these legacy devices. Further, the initial overhead of bootstrapping the system and generating the features for all existing content might be high. In this case, we can leverage some history of user access patterns to reduce the initial overhead. We can initially generate features/indexes only for commonly accessed content and grow the index as new content gets generated and accessed.

3.2 Efficient Search Mechanisms

Indexing Mechanisms.

Retrieving semantically related data inherently requires efficient *similarity* indexing and search mechanisms. For example, many of the feature generation algorithms described in the previous section use distance metrics like Hamming distance and Euclidean distance to find data that is semantically related. These distance metrics can also be combined with clustering mechanisms for better presentation and efficient retrieval [27]. We can leverage existing techniques such as locality sensitive hashing [11] to build efficient indexing and lookup mechanisms.

Cross-device search.

A natural question is how do we extend the above indexing and lookup mechanisms to search for data distributed across

different devices? Cross-device search should take into account the query workload and connectivity constraints of the devices. Strawman solutions—having every device broadcast queries to all devices or setting up pre-computed query routing paths—are simply not feasible in a heterogeneous home network environment.

We address these challenges using the following techniques:

- *Offloading proxies:* Instead of having a mobile device be in charge of querying and aggregating results across the network, we can choose a nearby less-constrained device (e.g., a laptop) to serve as a proxy. This proxy can query all neighbors, collect responses, and send only a filtered result stream to the user on the mobile device.

- *Replica-aware redirection and pruning:* The knowledge of replicas across devices can help reduce the search space. For example, if the pictures on a camera have been copied to a laptop, then the queries can be redirected to the laptop. If only some of the files have been copied, the search space on the camera can be reduced.

- *Leveraging user behavior:* We also use knowledge of user access patterns to prune the search space. As a simple example, if we know that users tend to search for recently added content, we can use this information to reduce the query overhead. Similarly, we can leverage cross-user similarity in search patterns to cache query results on well-provisioned devices.

- *Replicating indexes:* Search queries need not always be processed by the device where the content resides. Indexes for data on resource constrained or connectivity-restricted devices can be periodically replicated across less constrained devices. This can reduce the query processing load on mobile devices and also reduce query response time. Additionally, replicating the indexes can help locate related data on devices that may not always be connected.

3.3 Storage Management

Storage management is an important concern and it has two important facets: (1) managing limited storage space on devices efficiently, and, at the same time, (2) ensuring high data availability—by maintaining replicas of content—even when the devices are not always connected. Approaches for automatic storage management that meet these two requirements have been recently proposed in the context of home environments [18], but they are all forced to deal with exact copies of files.

In contrast, semantic features provide greater flexibility in storage management. In particular, users can express richer policies, such as, ensuring at least K copies of files, out of which M can be of lower resolutions, where the resolution is no lower than a certain threshold. This would help to manage space efficiently, since lower resolution files occupy less space. Users can also express other high level policies such as keeping at least K files related to some event. Such policies make the storage management task much simpler for users by automating it, and not requiring users to go through all files manually and finding which ones should be deleted for space efficiency. Finally, instead of deleting old files to conserve space, the file can simply be replaced by a lower resolution version and the index updated appropriately, thereby increasing the life-span of data.

4. RELATED WORK

We have discussed media fingerprinting techniques already in Section 3. Here we discuss other related work focusing on data management in home networks and coping with resource constrained devices.

Data discovery and synchronization.

Existing services like Bonjour [1] allow users to discover shared files across multiple personal computers. Media management software such as iTunes, iPhoto, etc., allow users to automatically synchronize data across portable devices. While they do provide a certain level of automation, these still require a significant amount of user effort. Further, these applications operate at the *data-level* and do not provide the rich semantic-aware capabilities that we envision.

Data replication.

Several research efforts have considered the problem of managing data across replicas. Many such approaches use version vectors to keep track of the most recent version for keeping replicas in sync [17, 20]. Recent systems such as PRACTI [8] and Cimbiosys [19] additionally provide partial replication capabilities to better utilize the available storage capabilities.

Perspective [23] provides users the capability to express their items of interest as "views" to simplify replica management. A view is similar to a search query for a group of files which share some attributes and a device they are stored on. Podbase [18] provides a framework for automatically ensuring that multiple copies are stored across devices. There are recent proposals that extend these schemes to take into account device capabilities and multiple representation formats [25, 26].

Again, these systems primarily operate at the data-level. Our approach can leverage many of the specific frameworks and algorithms these propose for maintaining data consistency, replication level etc. A key difference is that semantic-awareness brings additional flexibility to replica management (e.g., replica quality, related content).

Context-aware search.

The work that comes most close to ours in spirit is iScope [27] which provides a platform for users to search images in personal mobile devices. They use resource-aware search algorithms, similar to our proposal in Section 3. The key difference are: (1) we target an environment of heterogeneous devices in a home environment; (2) our framework can leverage devices with high compute power; and (3) we are interested in a broader notion of tracking semantic relationships that can be used for a wide spectrum of applications, beyond just searching for data.

Optimizations for mobile devices.

There are several previous and ongoing efforts for effectively utilizing the capabilities of mobile phones, focusing in particular on mechanisms for offloading compute-intensive tasks [7, 9, 10, 24]. These share common themes with our proposals for offloading indexing and query resolution functions.

5. DISCUSSION

In this section, we discuss other outstanding issues and potential extensions.

Access control.

Our description so far has assumed that any user within a household can issue a search for any piece of data. In practice, users are likely to place controls on who can access specific data (e.g., access controls for kids, visitors etc.). Along with semantic features, each file (or group of files) must also have an associated set of *permissions* which control whether a user's search for the file will receive a response. Furthermore, permissions associated with a file should be automatically and exactly transferred to all replicas and versions, since users apply controls based on the "information" contained in a file.

We note that our use of perceptual and contextual features provides an interesting mechanism for assigning permissions based on *content*. For example, using either perceptual fingerprints or contextual information a user may be able to associate a uniform set of policies for content that is considered objectionable for kids.

From a home to a community.

Our approaches for semantic relationships can also be used to manage information shared by a community of homes. Coupled with the appropriate access controls, semantic relationships can help users discover content in which they are all interested, e.g., photos of a community event, or even alternate versions of movies downloaded by a neighbor's set-top-box or media center. Of course a key challenge here is in facilitating queries across homes that may not be in radio range of each other. One way to do this is to leverage wired Internet connections of devices in the respective homes, or to use an external proxy to facilitate inter-home communication.

6. CONCLUSIONS

The goal of this paper is to look beyond traditional approaches for data management in home networks. In particular, we believe users' intent and requirements can be better captured by tracking semantic relationships between data. We propose leveraging recent developments in the multimedia processing literature for automatically inferring semantic features of multimedia content.

However, the decentralized, unmanaged, heterogeneous, and dynamic operational environment that characterize home users and devices present interesting systems and networking challenges. We have highlighted several candidate designs and system-optimizations that can help build an effective semantic-aware data management system for home networks. As ongoing work, we are creating a prototype implementation of a decentralized data management system.

7. REFERENCES

[1] Bonjour. `http://developer.apple.com/networking/bonjour/`.

[2] Intel study on home networks. `http://www.intelconsumerelectronics.com/Consumer-Electronics-3.0/Home-Storage-Architecture.aspx`.

[3] libFooID - free fingerprinting library. `http://www.foosic.org/`.

[4] Pandora. `http://www.pandora.com`.

[5] Shazam. `http://www.shazam.com`.

[6] AppleTV. `http://www.apple.com/appletv/specs.html`, 1999.

[7] R. K. Balan, M. Satyanarayanan, S. Y. Park, and T. Okoshi. Tactics-based remote execution for mobile computing. In *MobiSys '03: Proceedings of the 1st international conference on Mobile systems, applications and services*, pages 273–286, New York, NY, USA, 2003. ACM.

[8] N. M. Belaramani, M. Dahlin, L. Gao, A. Nayate, A. Venkataramani, P. Yalagandula, and J. Zheng. Practi replication. In *NSDI*, 2006.

[9] B. Chun and P. Maniatis. Augmented smartphone applications through clone cloud execution. In *HotOS XII: 12th Workshop on Hot Topics in Operating Systems*, 2009.

[10] E. Cuervo, A. Balasubramanian, D. ki Cho, A. Wolman, S. Saroiu, R. Chandra, and P. Bahl. Maui: Making smartphones last longer with code offload. In *MobiSys*, 2010.

[11] A. Gionis, P. Indyk, and R. Motwani. Similarity search in high dimensions via hashing. In *VLDB*, 1999.

[12] P. Indyk and N. Shivakumar. Finding pirated video sequences on the internet. Stanford Infolab Technical Report, Feb. 1999.

[13] C. E. Jacobs, A. Finkelstein, and D. H. S. Salesin. Fast multiresolution image querying. In *SIGGRAPH*, 1995.

[14] D. Lowe. Object recognition from local scale-invariant features. volume 2, pages 1150 –1157 vol.2, 1999.

[15] Q. Lv, M. Charikar, and K. Li. Image similarity search with compact data structures. In *CIKM*, 2004.

[16] A. Oliva and A. Torralba. Building the gist of a scene: the role of global image features in recognition. *Progress in brain research*, 2006.

[17] K. Petersen, M. Spreitzer, D. B. Terry, M. Theimer, and A. J. Demers. Flexible update propagation for weakly consistent replication. In *SOSP*, 1997.

[18] A. Post, P. Kuznetsov, and P. Druschel. Podbase: Transparent storage management for personal devices. In *IPTPS*, 2008.

[19] V. Ramasubramanian, T. L. Rodeheffer, D. B. Terry, M. Walraed-Sullivan, T. Wobber, C. C. Marshall, and A. Vahdat. Cimbiosis: A platform for content-based partial replication. In *NSDI*, 2009.

[20] D. Ratner, P. L. Reiher, and G. J. Popek. Roam: A scalable replication system for mobile computing. In *DEXA Workshop*, 1999.

[21] S.-C. Cheung and A. Zakhor. Estimation of web video multiplicity. In *Proc. SPIE–Internet Imaging*, 2000.

[22] B. Salmon, F. Hady, and J. Melican. Learning to share: a study of sharing among home storage devices. Technical Report CMU-PDL-07-107, Carnegie Melon University, 2007.

[23] B. Salmon, S. W. Schlosser, L. F. Cranor, and G. R. Ganger. Perspective: Semantic data management for the home. In *NSDI*, 2009.

[24] M. Satyanarayanan, P. Bahl, R. Cáceres, and N. Davies. The case for vm-based cloudlets in mobile computing. *IEEE Pervasive Computing*, 8(4), 2009.

[25] K. Veeraraghavan, J. Flinn, E. B. Nightingale, and B. Noble. quFiles: The Right File at the Right Time. In *Proc. FAST*, 2010.

[26] K. Veeraraghavan, V. Ramasubramanian, T. L. Rodeheffer, D. B. Terry, and T. Wobber. Fidelity-Aware Replication for Mobile Devices. In *Proc. Mobisys*, 2009.

[27] C. Zhu, K. Li, Q. Lv, L. Shang, and R. P. Dick. iscope: Personalized multi-modality image search for mobile devices. In *Mobisys*, 2009.