A Distributed Personal Content Management Architecture

Frédéric Iterbeke

Supervisor(s): Tim Wauters, Filip De Turck, Bart Dhoedt

I. Introduction

Since the introduction of the Web 2.0 there has been an enormous surge of user generated content. Users tend to have multiple accounts on sites such as Youtube, Flickr and the likes, which allow their users to manage their content and share it with each other. Moreover, with the mass production and consumption of media-capable mobile devices, this growing amount of personal content tends to get distributed even further among different devices and (online) user accounts. This increasingly complicates accessing and browsing this content, not to mention sharing it with other users. However, intuitively users tend to see their collections of personal content as a whole rather than a fragmented collection. Users thus need new, more efficient ways of searching, managing and sharing their personal files as well as browsing and/or downloading other users' files. This is the main goal of the Personal Content Management (PeCMan) system: to provide an open and distributed content management platform, offering content addressing and services in a flexible and transparent way. It aims to offer users a virtual compound drive on which all of a user's personal content can be found, shared, and, if necessary, stored. Key functionalities include (automatic) data tagging and tag-based browsing, federated identity management and user-centric security.

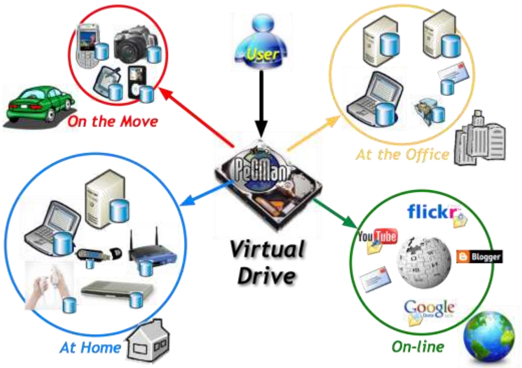

Figure 1. The PeCMan Virtual Drive Concept

II. Architecture

A centralized architecture has a single point of failure and can only scale up to a few thousand simultaneous user requests [1]. Therefore, a distributed architecture was needed to provide a robust and scalable platform for the PeCMan system. Moreover, the heterogeneity of devices taking part in the system supports a Peer-to-Peer (P2P) design. Unstructured P2P approaches proved unsatisfactory due to the lack of bandwidth scalability caused by their use of flooding algorithms, sending every message to every known peer. Thus, we developed a hybrid P2P architecture consisting of 2 tiers: peers (Nodes) and superpeers (Supernodes) [2]. Nodes have a direct connection to SuperNodes and typically, many Nodes will connect to each SuperNode. SuperNodes are interconnected by a Distributed HashTable (DHT), a structured P2P system that provides a distributed key-to-value mapping. DHTs organise themselves in

F. Iterbeke is with the Department of Information Technology (INTEC), Ghent University (UGent), Gent, Belgium. E-mail: Frederic.Iterbeke@intec.UGent.be .

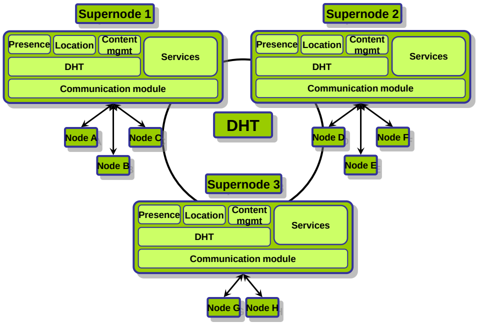

Figure 2. An overview of the distributed content management architecture.

a ring topology, providing lookup performance and structural maintenance logarithmically proportional to the number of DHT nodes in the network. These SuperNodes provide lookup capabilities on metadata, presence information and other services to the users of the PeCMan system via their Nodes. These services can be implemented independently from each other, so that changing one module does not require the architecture to change. Nodes communicate with SuperNodes through a communication module, which could comprise of multiple possible communication protocols. An overview of this architecture can be seen in figure 2.

III. Inner workings

For each user, one or more metadata files are kept on the DHT network. Whenever a user logs in via his preferred Node, this Node will provide a view on and means of interaction with this stored data. This is accomplished by mapping the username to the list of relevant metadata files. These files hold the mappings of user file references to their actual location, along with access rights and preferences. Note that the actual file location could be anywhere from the local hard drive of the current Node to an online-based storage provider such as YouTube or the flash drive from a mobile phone Node. The DHT stores this metadata redundantly through replication on different DHT nodes, so that no data is lost when a SuperNode goes offline. To support tag-based search, each tag is also mapped to a list of file references matching that tag. This enables users to search other users' (shared) content as well as their own in a user-friendly way. Since the amount of tags is potentially very large and many thousands of simultaneous requests should be possible, the performance of the system is very important. Users tend to be very conscious of latency; they lose interest if a query takes too long to complete. Therefore, improvements for service performance are being studied by deploying intelligent caching algorithms on the SuperNodes [3].

IV. Conclusions

In order to manage the explosive growth of personal content, a distributed network service is needed that offers access to personal content at any time, from anywhere and from any type of device, for multiple concurrent users. This article presents a distributed hybrid P2P architecture that can be used to achieve this goal. Ongoing research focuses on improving performance of the developed platform by deploying caching algorithms and measuring bandwidth, CPU and memory consumption on a large-scale testbed.

References

[1] P. Backx, T. Wauters, B. Dhoedt, and P. Demeester, "A comparison of peer-to-peer architectures," in *Proceedings of the Eurescom Summit 2002 Powerful Networks for Profitable Services*, 2002, p. 215222.

[2] F. Iterbeke, S. Melis, B. de Vleeschauwer, T. Wauters, F. De Turck, B. Dhoedt, P. Demeester, B. Theeten, and T. Pollet, "An open peer-to-peer based platform for scalable multimedia communication," in *The 2008 International Conference on Parallel and Distributed Processing Techniques and Applications*, 2008.

[3] N. Sluijs, T. Wauters, B. De Vleeschauwer, F. De Turck, B. Dhoedt, and P. Demeester, "Caching strategy for scalable lookup of personal content," in *Proceedings of The First International Conference on Advances in P2P Systems*, 2009.