A Scalable Web Hosting Platform using a Cloud and Peer-to-Peer Hybrid Architecture

CS854 Draft Project Proposal

Alexander Pokluda

Introduction

The pWeb project is a well established project aimed at creating a hybrid peer-to-peer and cloud-based platform for developing scalable web applications. Contemporary client-server web hosting applications deployed exclusively in the cloud face a number of challenges such as scalability, single point of failure, administrative overhead, and cost associated with maintaining a dedicated hosting infrastructure [1]. pWeb seeks to address these challenges by leveraging the strengths of both cloud and peer-to-peer infrastructures. This is accomplished by employing a three tier architecture. Tier I performs distributed name resolution and indexing of content hosted in the pWeb network. The infrastructure for this tier consists of nodes with longer uptimes and higher bandwidth than a typical node in a peer-to-peer network. It is expected that organizations, institutions and individuals will contribute heterogeneous cloud resources to this tier by hosting instances of the pWeb naming and routing software. This software will be based on the Plexus [2]. Plexus is a peer-to-peer search protocol that provides an efficient mechanism for advertising a bit sequence and discovering it using any subset of its 1-bits.

Content in pWeb is hosted from nodes in Tier II that participate in the peer-to-peer network. In general, these nodes are assumed to have a shorter but regular uptimes and less bandwidth. Nodes in this tier may be servers, desktop computers, laptops or event mobile devices. A user publishes content using the client software running on one of these nodes. The client software running on will push the content meta-data, such as content name and location, to the cloud hosted indexing and routing infrastructure in Tier I. The content itself will remain on the user's machine and be made publicly accessible. (In the initial version of pWeb, content will be served over HTTP, but as the system evolves, support may be added for other protocols). To ensure that content remains accessible despite peer churn and the limited resources of individual peers, the nodes in Tier II will be organized into replication groups. Each member of a replication group will mirror the content of the original publisher and make it publicly available as well.

Nodes in Tier III are consumers of content and use the public services offered by Tier I and Tier II to access it.

Significance

Other systems exist or have been proposed that share some similarities with pWeb; however, most of these systems are hybrid systems where a peer-to-peer network of caches is used to reduce the load on a traditional, centralized web server—the most notable is CoralCDN [3]. An exception is Freenet [4],

which provides a web hosting platform based on a distributed peer-to-peer data store. Frennet and pWeb have similar goals; however, they are based on fundamentally different architectures. Freenet used a flat peer-to-peer architecture that prioritizes anonymity and censorship resistance, and as a result, content can be slow to access and dynamic pages are not supported. pWeb used a unique three tier structured architecture that combines cloud and peer-to-peer resources. pWeb aims to democratize web hosting by providing individual publishers with the virtually limitless resources of a large peer-to-peer network with the stability and reliability of the cloud. pWeb will focus on providing support for dynamic content and an improved user experience with configurable security and privacy settings.

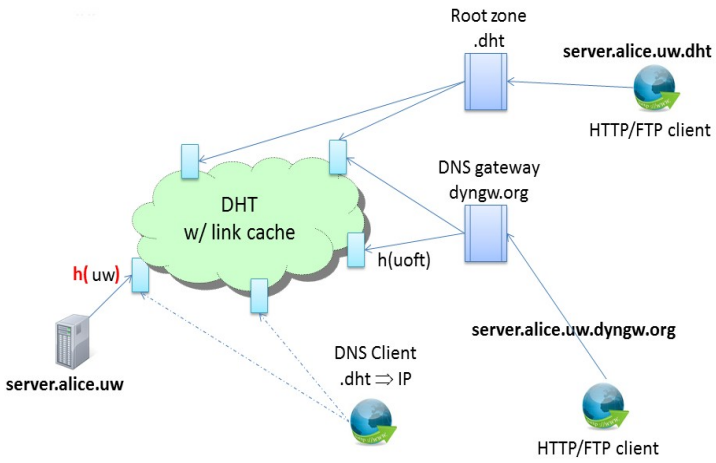
Related Work

Several related works were mentioned in the previous sections, including a paper giving a high-level overview of the pWeb system [1] and a paper describing in detail the Plexus peer-to-peer protocol and distributed hash table [2] that constitutes a core component of the pWeb system. Extensive work has been done to define the pWeb system and much of this work is available in progress reports submitted to Orange Labs, one of the project sponsors. The most relevant report is [5] that describes the architecture, design and implementation of pWeb.

Milestones

The work done for this project will focus on creating a working implementation of the pWeb system that can be demonstrated at the sponsor's facility in coordination with other project members. We will implement a working system using the three tier architecture and a top-down approach. The system will initially consist of the minimum set of features required with additional features being added incrementally in subsequent iterations.

Figure 1: Device name resolution in pWeb with DNS interface.



The initial version will not implement group formation and content replication. Devices in Tier II are assumed to be always online and have sufficient bandwidth to serve the published content. Tier I will allow users to resister and resolve device names in order to share and access content. Tier II devices will be able to register and update names using a name updater application that will function similar to a DDNS client and will serve content using a standalone HTTP server. Tier III devices will use a standard, unmodified web browser to access content on the pWeb network. Names will be resolved by the web browser using a DNS compatible interface provided by Tier I. This architecture is depicted in

Figure 1. Here content served by the device named server.alice.uw can be access by an unmodified web browser using the domain name server.alice.uw.dyngw.org, or the domain name server.alice.uw.dht if the DNS used has been configured to support the .dht pseudo top level domain.

Three categories of milestones are listed. The basic milestones serve as the baseline and include the minimal set of tasks that I expect to accomplish and will be my contribution to making the working pWeb system described here. The additional milestones are tasks that I hope to accomplish provided I do not encounter any difficulty meeting the basic milestones. The extra milestones are tasks that I will work on if the tasks for the basic and additional milestones are easier to complete than expected.

I. Basic Milestones

I will be responsible for creating the DNS interface to Tier I in order to enable unmodified Tier III devices to access content on the pWeb network. The primary task associated with this is programming a name server that receives DNS requests, performs a query for the name using Plexus, and returns the name using a DNS reply.

Faiz and Shihab are going to work on updating their implementation of Plexus that they created for the K-Links simulation to enable registering and updating device names. I will work with them to package their Plexus implementation as a library that can be integrated into the name server and name updater.

In my project report, I will provide a comparison of the scalability of Plexus and DNS as well as the amount of time required for name resolution.

II. Additional Milestones

If I am able to meet the basic milestones in a timely manner, I will work on creating a “name crawler.” The purpose of this name crawler will be to discover all names published in Plexus and create a web accessible directory of them. Traditional web search engines would then be able to index content on pWeb by first crawling the directory and then accessing the content published on pWeb. The end result will be a seamless integration between pWeb and traditionally hosted web content and a unified browsing experience for the end user.

III. Extra Milestones

If I am able to meet the basic and additional milestones in a timely manner, I will continue working on the unified Tier II pWeb software. This application will integrate the name updater with web server and include support for group formation and content replication. If it seems that I will have enough time to work on this aspect of the system, I will create a list of more specific tasks and milestones as the project progresses.

References

[1] Ahmed, R. and Boutaba, R. pWeb: P2P Web Hosting.

[2] Ahmed, R. and Boutaba, R. Plexus: A Scalable Peer-to-Peer Protocol Enabling Efficient Subset Search. Networking, IEEE/ACM Transactions on , vol.17, no.1, pp.130-143, Feb. 2009.

[3] Freedman, M., Freudenthal, E., and Mazières, D. (2004) Democratizing content publication with Coral. In Proc. 1st USENIX/ACM Symposium on Networked Systems Design and Implementation.

[4] Clarke, I., Sandberg, O., Wiley, B., and Hong, T. (2001) Freenet: A distributed anonymous information storage and retrieval system. Lecture Notes in Computer Science. 2009: 46-66.

[5] Admed, R., Bari, F., Boutaba, R., and Haque, R. (2011) Liverable-1.2: Architecture, Design and Implementation. Project report for Orange Labs.