##################################################

2014 - A Storage Service based on P2P Cloud System

##################################################

The drawbacks of the model in this paper are:

- It assumes a simple trust model which is in part controlled by "trusted" peers

- Trackers are responsible for receiving and routing all uploads from peers defeating the purpose of the P2P network!

- The model does not offer file fragmentation which means file replication is done on the file-level.

- The paper claims to provide confidentiality, availability, among other desired distributed system properties, but doesn't explain how it accomplishes them.

Quotes:

1. we can deduce that cloud computing is a model that allows access to files, applications or services in a ubiquitous and pervasive way through. It provides the illusion of unlimited and on-demand scalability.

2. Abstraction means that the implementation details of the system users and developers are abstracted.

3. Virtualization means that systems and storage can be provisioned as needed and resources are scalable with agility.

4. A distributed storage system is an infrastructure that allows to store files in nodes, which are connected through a computer network. These systems are characterized by their wide range of applications, that among most important are backup files, sharing of files in network and edition of documents from different locations.

##################################################

2008 - Prosa P2P Resource Organisation by Social Acquaintances

##################################################

Quotes:

1. Small–world network presents both small network diameter and high clustering degree

(i.e., respectively, the maximum distance, in number of hops, between two generic nodes of the network && good connections among similar or related nodes)

##################################################

2009 - A Cloud Computing Platform Based on P2P

##################################################

The main advantages of this model are:

- It removes the central point of failure that exists in the Google File System, by making Chunk Servers part of a DHT Chord like structure.

- Chunk Servers will not only contain data blocks and respective versioning, like in GFS, but also the index module which is based on DHT arithmetic like in structured P2P overlays and a routing module that passes lookup requests by a next-hop routing table also assigned by DHT.

- File updates are done on a single replica using a distributed-lock-like-mechanism, managed by the index server, who figures if concurrent writes are being done. If they aren't, the client is given the chunk server with the highest version, to which the Client writes the updates. When the client finishes writing, he informs the index-server, who starts a synchronization process on all Chunk servers.

The drawbacks of the model in this paper are:

- Although chunk servers are organized in a P2P fashion, only one chunk server responds to the worker with the data.

- The P2P search request package routed among the chunk servers following the P2P search protocol such as Chord, eventually, the request reaches the server which contain the index information of the logic ID in searching. The index includes all the pointers of the data replica with the same ID. The chunk server now acts as an index server (Master Server in traditional GFS). The chunk server will select a latest pointer by its version number and by chunk server proximity and return it to the client. Client App gets the best address, it will then send its request to the address of the chunk server which contains the data block

##################################################

Cloud@Home: Bridging the Gap between Volunteer and Cloud Computing

##################################################

The main advantages of this model are:

- Architecture is divided in 3 modules, Frontend Layer (Server/Client Apps), Virtualization Layer and Physical Layer.

- Client App of the front-end handles authentication and interaction mechanisms

- Server App of the front end allows clients to make their resources available or search for resources, with specific requirements such as QoS, availability, etc.

- Virtualization layer manages two services... execution services and storage services.

- The execution service is the tool provided by the virtual layer for creating and managing virtual machines. A user, sharing his/her resources within a Cloud@Home, allows the other users of the Cloud to execute and manage virtual machines locally at his/her node, according to policies and constraints negotiated and monitored at the frontend. Virtual machine nature provides isolation and thus security.

- The storage service implements a storage system distributed across the storage hardware resources composing the Cloud, highly independent of them since data and files are replicated according to QoS policies and requirements to be satisfied. From the end-user point of view, a storage Cloud appears as a locally mounted remote disk, similarly to a Network File System or a Network Storage.

The disadvantages of this model are:

- Clients pay to allocate data on some other client resources, but apart from availability and QoS there seem to be no replication guarantees. One can assume that the "hosting" client provides some sort of replication, but the protocol doesn’t specify its enforcement.

- The Architecture is a typical client server one, no P2P benefits exist.

##################################################

Design and Implementation of a P2P Cloud System - P2P Cloud System

##################################################

Acknowledges the fact that P2P clouds are unlikely to provide the same QoS guarantees that centralized or federated clouds do, but that scenarios exist where they can still be beneficial.

Is focused on infrastructure as a Service, IaaS... Virtualization, Storage, etc...

advantages:

- Client Side Daemon software handles cohesion and churn gracefully for the p2p cloud network.

- Clients are not required to be reliable, they can join or leave the network at any time.

- Isn't limited to storage services only.

disadvantages:

- No QoS guarantees are provided since each node in the P2P network are managed by the respective owners.

Quotes:

1. P2P Cloud can be assembled at virtually no cost using existing resources; therefore, many small or medium-sized organizations could turn idle resources into a computing infrastructure which can be partitioned among a number of internal customers".

2. A P2P Cloud would provide on-demand scalability, access to computing and storage space with no single point of failure nor central management.

##################################################

P2P and Cloud: A Marriage of Convenience for Replica Management

##################################################

Their goal is to build an unstructured overlay linking the replicas of a data object and maintaining a minimal size despite churn. For that they use a few well-known protocols like CLOUDCAST [11] and CYCLON [17] for overlay maintenance and an aggregation protocol [7] to monitor the overlay size.

We consider the following three thresholds for the overlay size: (1) redundant R,

(2) sufficient S and (3) critical C, with R > S > C.

In general, the size of the overlay is expected to stay in the range [S,R]. When the current size is larger than R, some peers could be safely removed; when it is smaller than S, some peers (if available) should be added. When it is smaller than C, the system is in a dangerous condition and part of the service should be provided by the cloud.

similarities:

- replicas containing the same data objects are organized in their own overlays. If P2P nodes aren't enough, regular cloud is used for a limited time.

advantages:

- When peers are too low, storage can be done levering existing cloud servers!

Quotes:

1. Companies like Spotify and Wuala have started to explore how the two worlds could be merged, exploiting

2. The lack of centralization provides scalability, while exploitation of user resources reduces the service cost. However, several drawbacks exist, like availability and reliability.

3. While P2P remains a valid solution for cost-free services, the superior QoS capabilities of the cloud makes it more suitable for those who want to create novel web businesses and cannot afford to lose clients due to the best-effort philosophy of P2P.

4. The problem to be solved is the maintenance of an appropriate redundancy level in replicated storage. Consider a system where a collection of data objects is replicated in a collection of peers. A data object is available if at least one replica can be accessed at any time and is durable if it can survive failures.

5. One of the most important limitations in P2P-based storage systems relates to the difficulty to guarantee data reliability. P2P nodes leave or fail without notification and stored data can be temporarily or permanently unavailable. Replication, Erasure coding or a combination of both can be used to increase reliability.

6. Two main approaches exist for replica control: (1) proactive and (2) reactive. Proactive approach creates replicas at some fixed rate [14,13], whereas in reactive approach a new replica is created each time an existing replica fails [9,3].

#########################################

CYCLON: Inexpensive Membership Management for Unstructured P2P Overlays

#########################################

1. Ideally, the resulting overlays should have low diameter, be resilient to massive node failure and resistant to churn.

2. Gossiping networks generally exhibit self-healing behavior with respect to major network disasters. It’s also been observed that some of these overlays exhibit small-world and scale-free networks [3,4], however in many cases it is better to construct overlays that are close to random graphs [5].

3. CYCLON introduces an enhanced version of shuffling protocols which result in node-degree distributions that exhibit better properties than those found in overlays resulting from basic shuffling [7] or even random graphs. It is also more efficient and provides better quality Management of node additions and removals. Thus, introducing an expensive Membership management that does not disrupt randomness of the overlay networks.

4. Degree distribution is highly related to the robustness of the overlay in the presence of failures as it shows the existence of weekly connected nodes and massively connected hubs. Degree distribution is also a good indication of the way epidemics are spread. Finally, Degree distribution also provides indication of how fairly links are distributed among nodes and therefore an indication of the distribution of resource usage across nodes.

5. For the sake of robustness, efficient information dissemination and load balancing, it is important to have a balanced, uniformly distribution of links across all nodes of the overlay, i.e. we want to have a degree distribution with low standard deviation.

6. New nodes joining the network contact an introducer I, I initiates c (cache-size) random walks, some of them may fail, to introduce the joiner J to other nodes in the network and introduce those nodes to J.

7. When a node leaves the network for any reason, timely elimination of dead links is crucial for robustness.

8. CYCLON doesn't use pings or heartbeats for failure detection. Instead they leverage shuffling mechanisms. When a peer P initiates a shuffling with a peer Q, if it doesn't obtain an answer within a predefined time it assumes it died and removes Its entry from the neighbor’s cache. Leading to the removal of deadlinks from the network overtime. Timeout should be at least twice if the typical latency of the underlying network. CYCLON assumes that newer nodes, with less age, have higher probability of being alive. Such young nodes are also not usually selected for shuffling. Because older node links are gradually replaced overtime, if they disconnect, their links don't linger around in the network.

9. CYCLON bandwidth consumption depends on the amount of transferred data per cycle and the cycle duration.

10. CYCLON uses UDP Packets.

#########################################

Cloudy Weather for P2P, with a Chance of Gossip

#########################################

Quotes

1. While executing active servers on elastic computing facilities like Amazon EC2 and pairing them with user-provided

peers are definitely one way to go, this paper proposes a novel approach that further reduces the economic cost.

Here, a passive storage service like Amazon S3 is exploited not only to distribute content to clients, but also to build and manage the P2P network linking them. An effort is made to guarantee that the read/write load imposed on the storage remains constant, regardless of the number of peers/clients. These two choices allow us to keep the monetary cost of the cloud always under control, in the presence of just one peer or with a million of them. We show the feasibility of our approach by discussing two cases studies for content distribution: the Dilbert’s comic strips and the hourly News Update podcast from CNN.

2. while you cannot beat P2P from an economic point of view, the superior availability of the cloud (in the order

of 99.99%) makes it a more believable environment for those who want to create novel web businesses and cannot afford to lose clients due to the best-effort philosophy of P2P [1].

3. Note, however, that CYCLON cannot be used without modifications, for various reasons. The first reason is that since the cloud is a key/value store, it cannot perform autonomous computations; hence, a peer p interacting with the cloud must play the cloud role as well. A second reason is that the unmodified CYCLON protocol does not work well when only few peers are active in the system. Moreover, the number of cloud descriptors risk to be too small