An Efficient Caching Scheme and Consistency Maintenance in Hybrid P2P System

Abstract:

Peer-to-peer overlay networks are widely used in distributed systems. P2P networks can be divided into two categories: structured peer-to-peer networks in which peers are connected by a regular topology, and unstructured peer-to-peer networks in which the topology is arbitrary. The objective of this work is to design a hybrid peer-to-peer system for distributed data sharing which combines the advantages of both types of Peer-to-peer networks and minimizes their disadvantages. Consistency maintenance is propagating the updates from a primary file to its replica. Adaptive consistency maintenance algorithm (ACMA) maintains that periodically polls the file owner to update the file due to minimum number of replicas consistency overhead is very low. Top Caching (TC) algorithm helps to boost the system performance and to build a fully distributed cache for most popular information. Our caching scheme can deliver lower query delay, better load balance and higher cache hit ratios. It effectively relieves the over-caching problems for the most popular objects.

Existing System

In the past decade a number of prototype peer-to-peer information retrieval systems have been developed. Unfortunately, none of these has seen widespread real-world adoption and thus, in contrast with file sharing, information retrieval is still dominated by centralized solutions. In this

article we provide an overview of the key challenges for peer-to-peer information retrieval and the work done so far. We want to stimulate and inspire further research to overcome these challenges. This will open the door to the development and large-scale deployment of real-world peer-to-peer information retrieval systems that rival existing centralized client-server solutions in terms of scalability, performance, user satisfaction, and freedom.

The proposed hybrid peer-to-peer system is composed of two parts: the first part is a structured core network which forms the backbone of the hybrid system; the second part is made of multiple unstructured peer-to-peer networks each of which is attached to a node in the core network. The core structured network can narrow down the data lookup within a certain unstructured network accurately, while the unstructured networks provide a low-cost mechanism for peers to join or leave the system freely.

Proposed System

we propose a hybrid peer-to-peer system for distributed data sharing which combines the structured and unstructured peer-to-peer networks. In the proposed hybrid system, a structured ring-based core network forms the backbone of the system and multiple unstructured peer to peer networks are attached to the backbone and communicate with each other through the backbone. The core-structured network provides an accurate way to narrow down the queried data within a certain unstructured network, while the unstructured networks provide a low cost mechanism for peers to join or leave the system freely.

The main contributions of this paper can be summarized as follows:

- Propose a hybrid peer-to-peer system for distributed data sharing. It utilizes both the efficiency of the structured peer-to-peer network and the flexibility of the unstructured peer-to-peer network, and achieves a good balance between the efficiency and flexibility.
- To maintain consistency, using file consistency algorithm for hybrid P2P system so that periodically the file owner to update the file due to number of replicas consistency overhead is very low.
- To boost the performance of hybrid P2P, Top Caching (TCS) algorithm is used to build a fully distributed cache for popular information in P2P systems. It effectively relieves the over caching problems for the most popular objects.

IMPLEMENTATION

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

The implementation stage involves careful planning, investigation of the existing system and it’s constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

Main Modules:-

Algorithm used Adaptive consistency maintenance algorithm (ACMA)

Modules:

1. Core Transit Network:

The core transit network, called t-network, is a structured peer-to-peer network, which organizes peers into a ring. We call peers in the t-network t-peers. Each t-peer is assigned a peer ID. Each t-peer maintains two pointers, which point to its successor and predecessor, respectively.

2. Stub Network:

A stub network, called s-network, is a Gnutella-style unstructured peer-to-peer network. The topology of an s-network is arbitrarily formed. Each s-network is attached to a t-peer and this t-peer belongs to both the t-network and the s-network. One thing to mention about the s-network is that the topology of an s-network is a tree instead of a mesh.

3. Idea of Hybrid Peer:

The basic idea behind the hybrid peer-to-peer system is that the t-network is used to provide efficient and accurate service while the s-network is used to provide approximate best-effort service to accommodate flexibility. Peers can join either t-network or s-network directly. The hybrid

system can effectively reduce the topology maintenance overhead caused by peer joining or leaving.

4. Consistency Algorithm:

In the distributed data sharing, the consistency of the data needs to be focused because there are two different networks are built on single. Maintaining consistency between frequently updated or even infrequently updated files and their replicas is a fundamental reliability requirement for a P2P system. P2P systems are characterized by dynamism, in which node join and leave continuously and rapidly. Moreover, replica nodes are dynamically and continuously created and deleted. For consistency maintenance, we introduce an algorithm for hybrid network, which is known as Adaptive File Consistency Algorithm (AFCA).

System Configuration:-

H/W System Configuration:-

Processor	-	**Pentium –III**
Speed	-	1.1 Ghz
RAM	-	256 MB(min)
Hard Disk	-	20 GB
Floppy Drive	-	1.44 MB
Key Board	-	Standard Windows Keyboard
Mouse	-	Two or Three Button Mouse

Monitor - SVGA

S/W System Configuration:-

- Operating System :Windows95/98/2000/XP
- Front End : java, jdk1.6