Multiple Super Peers to Reduce Single Failure of Super Peer

I Wayan S Wicaksana

iwayan@staff.gunadarma.ac.id

Abstract:

Recently, Peer-To-Peer (P2P) has become popular media to share data and resource. A new wave of P2P systems is advancing an architecture of centralized topology embedded in decentralized systems; such topology forms a super-peer network. At this paper, we purpose an approach with multiple of super peers model to reduce single point failure of super peers.

1. Introduction

Peer-to-peer (P2P) systems are designed to distribute functionality and resources among a large number of independent hosts. The promise of this design is that highly distributed state is easier to scale, is more resilient to failure and supports greater administrative autonomy [Alper Tugay Mizrak and Savage, 2003]. However, the costs of distribution can be significant as well because of the happening of join or leave peer into network when cannot guess.

By introducing the concept of super peer, the topology of P2P is now organized through a two-level hierarchy: nodes that are faster and/or more reliable than normal nodes take on server-like responsibilities and provide services to a set of clients. The super peer paradigm allows decentralized networks to run more efficiently by exploiting heterogeneity and distributing load to machines that can handle the cost. On the other hand, it does not inherit the flaws of the client-server model, as it allows multiple, separate points of failure, increasing the health of the P2P network [Montresor, 2004]. The usefulness of the super peer approach is not limited to file-sharing. For example, it is possible to envisage distributed game systems [J. Smed and Hakonen., 2003], where multiple locations of a simulated virtual environment are maintained by a distributed collection of super peers, that manage the virtual environment on behalf of their clients. Grid management systems [Foster and Kesselman., 1999] and distributed storages [et al, 2000] are other possible candidates for such architectural paradigm.

Building and maintaining a super peer topology is not simple. The extreme scale and dynamism call for robust and efficient protocols, capable to self-organize and self-repair a super peer overlay in spite of both voluntary and unexpected events like joins, and leaves. Other side also how to choose or determine super peer, and how to decrease risk of failure from super peer. So, the problem is complicated by the fact that the shape of a super peer topology is strongly application-dependent. Several questions arise: How many super peers are needed? How super peers are linked together?

Contributed of this paper is to evaluate multi super peer in overcoming single failure simulation, like seeing how many super peer which required in a cluster, how many cost for synchronization between super peer in a cluster.

This paper will be divided as follows, first section give problems and background of P2P in super peer. Second section elaborate elementary architecture from various configurations of P2P. The third part will evaluate super peer requirement in a cluster. And last part is conclusion of the paper.

2. Theory Background

Centralized System Centralized systems form the most popular system topology, typically seen as the client/server pattern. All function and information is centralized on a single server (sometimes referred to as the hub), with many clients connecting directly to the server to send and receive information. Both control flow and data flow take place through the central server. The primary advantage of centralized systems is their simplicity. Because all data is concentrated in one place, centralized systems are easily managed and have no questions of data consistency or coherence. Centralized systems are also relatively easy to secure, since there is only one host to be protected.

Pure P2P The main point of pure P2P is the skill of scalability, because peer can be joining with the network and begins to exchange data to another peer as soon as possible. However, the decentralized system had a fault - tolerance problem too, like a shutdown of any kind of peer, but that was not given the serious effect to system for generally.

Hybrid P2P In a hybrid peer-to-peer system, the control information is exchanged through a central server, while data flow takes place in a pure peer-to-peer manner as above. This architecture alleviates the manageability problems of pure P2P systems. The control server acts as a monitoring agent for all the other peers and ensures information coherence.

Super Peer P2P A new wave of peer-to-peer systems is advancing architecture of centralized topology embedded in decentralized systems; such topology forms a super-peer network. The advantages if we use super peer are:

- **Reduced time & bandwidth for search** Searching process is faster than the other system, into a smaller set of super peer with the peer element. Some of the examples of flooding in the pure/hybrid P2P will needed $O(N)$, while for The super peer will needed $O(N/M)$, where M is the amount of peer in a super peer.
- **Autonomous Units** Every super peer will be an autonomy unit, it is not depend on central server.
- **Manageability** Super peer will be more reliable in monitoring the peer element.
- **Load Balancing** In pure P2P every peer will has a same position and responsibility to network capability and computation. It can make easily decreasing of demonstration in work cause the network fragmentation as a effect of peer capability that not enough. The problem can handle with super peer, because only powerful computer will had status be super peer.

Although super peer will be more efficient, scalability, makes easily to managing, but super is one kind of single point of failure for the peer member. The one of approach is make multiple super peer in a cluster. The paper will show about how many cost and super peer needed.

3. Evaluation Methodology

3.1. Evaluation Judgment

There are some specific characteristics that make the P2P system different form other systems. Some papers said the characteristics are goal as well for the P2P system. The main characteristics of P2P system are:

1. Self-organizing, node has freedom to organize himself into network.
2. Symmetric communication, node are equal both request or offer services. So, nodes need have an operational computer of server quality, since each node can be act as a client and/or a server
3. Decentralized, no global directory or central control to every node.

From above main characteristic, there are other derivative characteristics as follow:

1. Autonomy, this character derives from self-organizing. A peer can consider himself to decide P2P model, joint to which P2P network and so on.
2. Cost of ownership, this characteristic come from understanding of ownership, shared ownership reduce cost of owning, and cost of maintaining. P2P systems implement this understanding

3. Anonymity / privacy, some forms of anonymity from the authors of the Free Haven are:
    - The location of a file is not known by its retriever
    - Files move freely among systems
    - Author / creator / publisher / reader cannot be identified
    - Servers do not know what documents they are storing
    - Servers cannot tell what document it is using to respond to a user's query
4. Scalability, decentralization gives better scalability, because limitation of scalability depend on power of centralized operation.
5. Ad-Hoc Connectivity , ideally connectivity can be variety, so peer can join and leave based on physical location or interest.
6. Addressing system is independent of the DNS
7. Joint and leave P2P nodes are at any time and unpredictable

The evaluation is to see how much super peer that needed in a cluster, and to see how to much the cost of handling join and leave between peers. A simulation is conducted to evaluate advantages and cost by implementing of multiple super peer.

Some requirements are needed to run P2P systems based on the characteristics, the requirements are:

1. Standard communication protocols are required
2. Information exchange should be secure
3. Information network should support policy-based authorization
4. Information network should facilitate more effective search
5. Information network should be easy to use and set up
6. Information network should scale
7. Information network should be ubiquitous

P2P system is proposed to answer some problem at many areas, P2P systems can be implemented for following examples:

- Community Web Network, can share data, information, sources among community group which has specific interest.
- e-Business, P2P has possibility to add new capability at distributing and sharing information
- Gaming, refer to messenger application, game has the same model, but need of

bandwidth is much more

- Collaboration development, such as rendering graphics.

Super Peer Model

Model is based on pure P2P, but some nodes will be signed as super-peer or super-node than rest of the nodes. Some nodes act as Super-Peers because their capacity, connectivity or reliability. A Super-Peer can keep a list of connection nodes and speed up the join process. To select super node can be automatically or manual. In the implementation if a super-node down, nodes can select other super-node. If there is no more super-nodes, node can act as super-node for himself. Example is Kazaa, and recent Gnutella shift to this model.

Figure 1 as example of network, P0 sent query to super-peer SP1, SP1 will check at his member (P0 and P1), is there the answer? If no the query will route to other super-peer (SP2, SP3, and SP4) not directly to peer (P2, P3, and P4). This approach can reduce traffic and speed of searching.

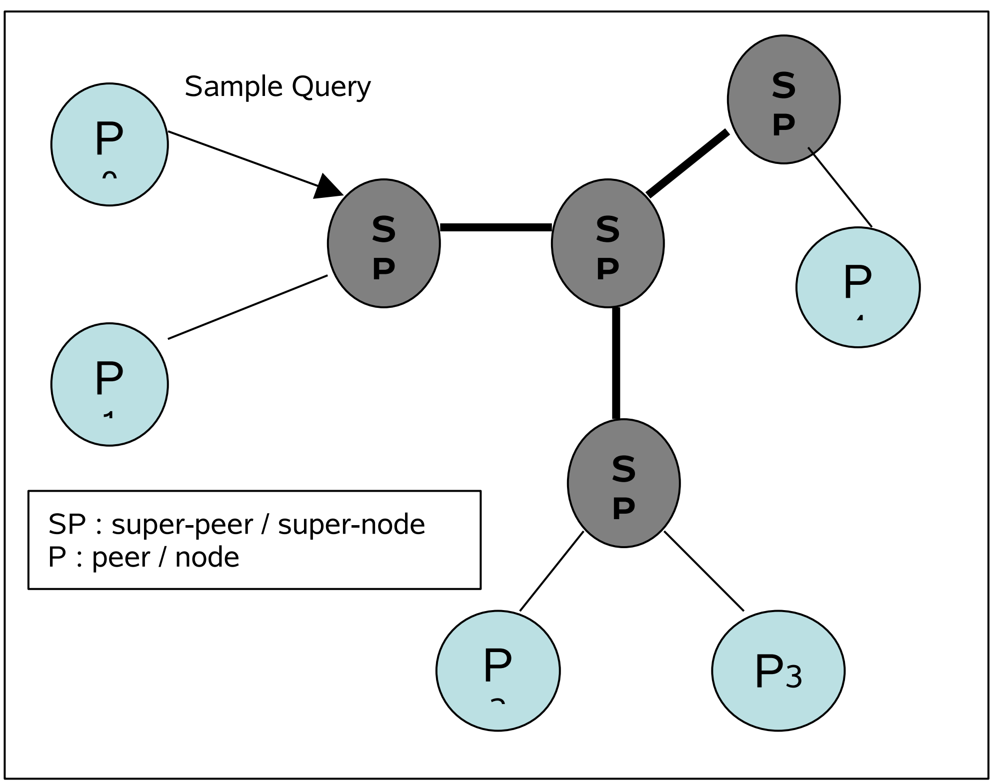

Figure 1. Partially P2P or Super-Peer Model

Improvement is:

- Super-Peer-Based routing and clustering strategy with RDF-based

3.2. Parameter and Evaluation Scenario

- For evaluated how many super peer that needed, will do a simulation that give a random value for failure from a peer (MTBF - Mean Time Between Failure), and if there are problems, how much time that needed to repair it (MTBR - Mean Time Between Repair). This simulation will happen for 100 times, and every simulation is modeled for 5 years (1 day is modeled in 1 second).
- For evaluating maximum join cost and leave cost, parameters are included for generate the random value from the amount of peer that join and leave with the cost join and leave. Simulation is modeled for 25 days.

  The formula for calculating join cost is:

$$JoinCost = \eta * \Phi * \varsigma \tag{1}$$

  where: η=number of join peers; Φ= size of kB for register cost per peer; ς=number of super peer in a cluster.

  The formula for calculating leave cost is:

$$LeaveCost = \Psi * \Theta * \varsigma \tag{2}$$

  where: Ψ=number of leave peers; Θ= size of kB for un-register cost per peer; ς=number of super peer in a cluster.

Parameters that is used for the simulation is shown at table 1. Simulation was ran at processor Intel 1.66GHz by memory 512MB by Microsoft Windows XP Professional. So that the result of simulation will very depend on machine and operating system and application which are used.

Table 1: Setup Value for Simulation

Information	Value	Unit
Num of peer Join	0-10	p/25day
Num of peer Leave	0-3	p/25day
Register cost	1000	bytes

Broadcast cost	1000	bytes

4. Result and Discussion

Result of simulation is showing at following table with the following details:

- Evaluation for the needs of super peers amount at table 2.
- Evaluation for cost of join and leave bandwidth on cluster at table 3.

Table 2: Failure of Super Peers

Scenario	Setup			Total of	Result (all failure)			
	value	min	max	super peer(s)	all failure	average	min	max
1	MTBF	50	200	1	100 times	46 days	30 days	61 days
	MTBR	1	5	2	14 times	1 days	0 day	4 days
				3	0 time	0 day	0 day	0 day
2	MTBF	70	150	1	100 times	60 days	40 days	86 days
	MTBR	1	7	2	40 times	1 days	0 day	8 days
				3	0 time	0 day	0 day	0 day
3	MTBF	10	250	1	100 times	30 days	22 days	39 days
	MTBR	1	3	2	6 times	1 days	0 day	3 days
				3	0 time	0 day	0 day	0 day

Table 3: Join and Leave Cost

Join Cost (kB/25 days)	Leave Cost (kB/25 days)
30,000	9,000

From simulation tables above, we can analyze that:

- Related to table 2, with 2 super peers, the network have owned the adequate ability, and with 3 super peers from simulation result 100 times, that from super peer will always available.
- Related to table 3 that the maximum join and leave cost every 25 days are not big traffic refer to current network bandwidth.
- Result of simulation can be considered that multi super peer can reduce failure of single peer

with cost of traffic is very low. The main problem in real implementation is how to choose backup super peer and how to create a protocol of changing the super peer.

5. Conclusion

Super peer is an effectively network for P2P, because it merges the centralized model and the pure P2P. This is to get the superior from both topology model and eliminate its weaknesses mutually.

The problem of single point of failure from super peer can be decrease by using multiple super peers. From the evaluation based on simulation, the result is that by using 3 super peers has been appropriate to solve this problem. In other side, the cost of network is relatively very small for synchronization between some super peers in a cluster.

Even though from simulation has been shown interesting result, but it need to be tested in a prototype which mimic real condition as open, autonomy, and dynamic condition. For the future work, we will develop prototype to test the multiple super peers model

Bibliography

Alper Tugay Mizrak and Savage, 2003

Alper Tugay Mizrak, Yuchung Cheng, V. K. and Savage, S. (2003). Structured superpeers: Leveraging heterogeneity to provide constant-time lookup. In *The IEEE Workshop on Internet Application*, page 104-111, SanJose, CA, USA.

et al, 2000

et al, J. K. (2000). Oceanstore: An architecture for global-scale persistent storage. In *the 9th Int. Conf. on Architectural Support for Programming Languages and Operating Systems*. Cambridge, MA.

Foster and Kesselman., 1999

Foster, I. and Kesselman., C. (1999). *The Grid: Blueprint for a Future Computing Infrastructure.* Morgan Kaufmann,.

J. Smed and Hakonen., 2003

J. Smed, T. K. and Hakonen., H. (2003). Networking and multiplayer computer games-the story so far. *International Journal of Intelligent Games & Simulation*, 2.

Montresor, 2004

Montresor, A. (2004). *A Robust Protocol for Building Superpeer Overlay Topologies*. Technical Report UBLCS-2004-8, Dept of Computer Science, Univ. of Bologna, Mura Anteo Zambony 7, 40127 Bologna,