# *A Skipnet Implementation for the Peersim Simulator*

A project by

Rodolfo Cartas Ayala
rodolfocartas@gmail.com

Submitted to

Dr. Alberto Montresor

March 28, 2007

# A Skipnet Implementation for the Peersim Simulator

Rodolfo Cartas
rodolfocartas@gmail.com

## *Introduction*

Skipnet is a P2P protocol that constructs a Dynamic Hash Table (DHT) with two particular properties[1]: Content Locality and Path Locality. Content locality refers to either place data on specific overlay nodes or distribute it across nodes within an organization. Path locality refers to the ability to guarantee that message traffic between two overlay nodes within the same organization are routed within that organization only. These properties overcome many disadvantages of other DHT building protocols that have no control over where data is stored (uniformity of the DHT) and its difficult to keep the data traffic within a organizational domain.

These properties are achieved through the use of a numeric address space and a string name space. The numeric address space guarantees that nodes are randomly distributed whereas the string name address space allows alphabetical sorting. Each node stores a number of pointers that skip nodes of the address spaces exponentially

## *Basic Skipnet*

The protocol intends to maintain a sorted ring of nodes organized alphabetically and pointers that skip over increasing number of nodes. Each skipnet node stores $2\log N$ pointers, where N is the number of nodes in the system. This table indicates nodes that are $2^h$ nodes away to left and right, where h is a level of the routing table making possible to route a message in $o(\log N)$ hops with high probability. A full skipnet consisting of 8 nodes is shown in Figure 1. Each ring at level h is split in two rings at leve h+1 by having each node randomly and uniformly choose to which of the two nodes it belogs.

### Routing

Routing by name in SkipNet is done following pointers that route closest to the intended destination. At each node, the pointer that points the furthest without pointing past the destination is used to route the message.

By using an order preserving hash function to generate numeric IDs, messages can be routed in the same way as with the string name version. Looking at the number ID in binary form determines the skip list style pointers of the node. The skip list style pointers of a node point to other nodes with the same numeric ID prefix.

---

[1] HARVEY, N. J. A., JONES, M. B., SAROIU, S., THEIMER, M., AND WOLMAN, A. Skipnet: A scalable overlay network with practical locality properties. In Proceedings of USITS (Seattle, WA, March 2003), USENIX. http://citeseer.ist.psu.edu/harvey03skipnet.html

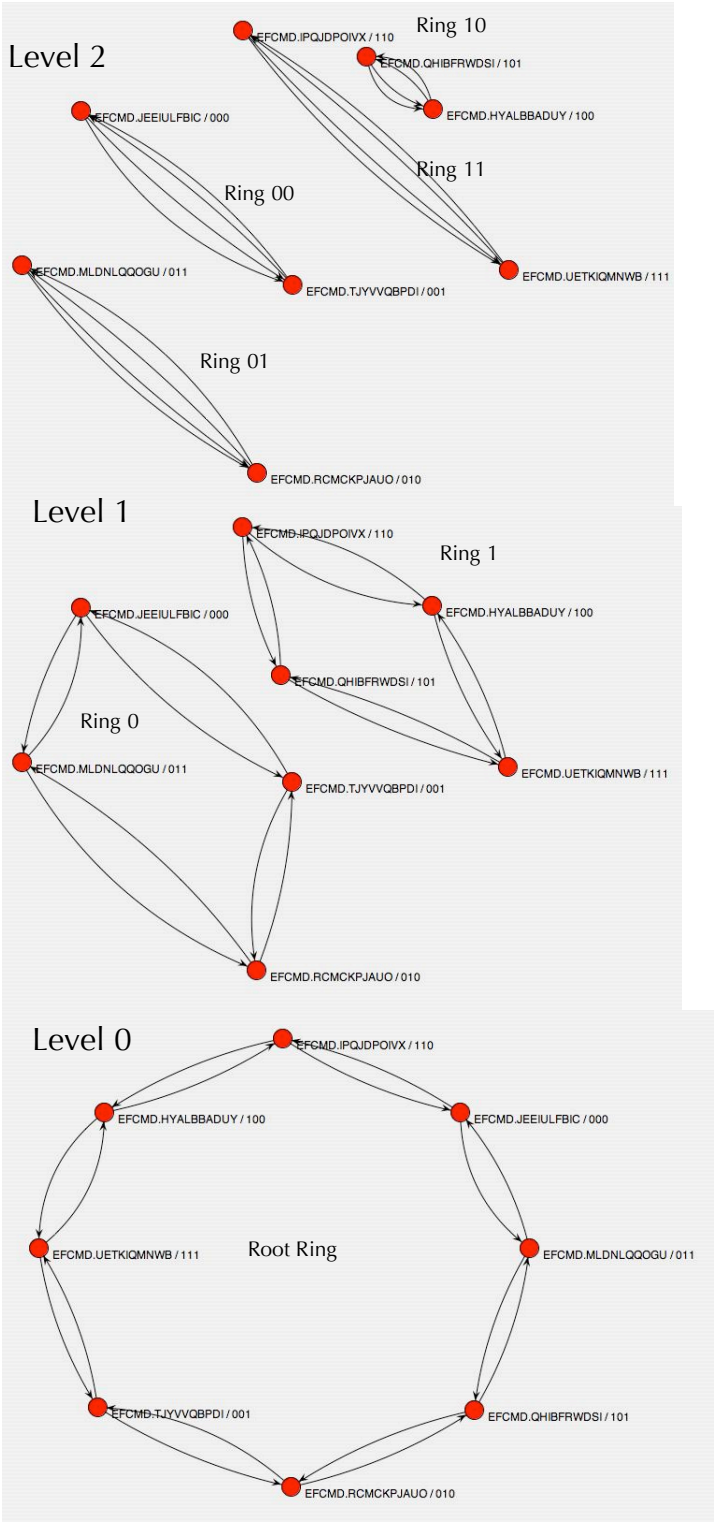


**Figure 1. Routing scheme for an 8 nodes skipnet.**

### Joining nodes

To join the SkipNet, a node must find the top-level pointer ring to which it belongs. This is basically equivalent to routing a message to the new node's numeric ID. Once it has found the top-level ring to which it belongs, the new node can find its neighbors by ordering its string name ID. The string name ordering process is then repeated for all lower level rings until the root is reached.

## *Implementation and Evaluation*

The protocol was implemented with the latest version of the Peersim Simulator (as of January 2007) following the event-driven model. Only the basic skipnet protocol was implemented. The skipnet protocol with proximity routing table is out of the scope of this exercise. In order to test the implementation, two benchmarks were performed: 1) Average hop count; and, 2) Number of clusters after node removal in a static skipnet. In order to obtain these measurements, each run is divided in connection, heat-up, and measuring periods. Measurements were conducted in networks growing from 128 nodes to 2^16 nodes in size.

### Average hop count

The goal of this exercise was to analyze the routing efficiency of the protocol once the DHT is built. The measurements obtained were the average hop count, its standard deviation, and the maximum hop count. For each network size, a number of 10000 messages were sent from random sources to random destinations using both numeric and name routing methods.

Results obtained from these simulations are depicted in Table 1 and Figure 1. As it can be seen, the skipnet protocols offers good small-properties having a similar average hop count to the one expected in case of numeric routing and smaller in case of name routing. This can be explained due to the use of a leaf set used to store the closest neighbors in the root ring, thus avoiding a few hops when a packet approaches its destination.

| Network size | avg(h) name | stdev(h) name | max(h) name | avg(h) numeric | stdev(h) numeric | max(h) numeric |
|---|---|---|---|---|---|---|
| 128 | 2.7876 | 1.5696 | 12 | 6.1102 | 3.1269 | 20 |
| 256 | 3.6247 | 1.9050 | 14 | 6.7520 | 3.1801 | 23 |
| 512 | 4.5080 | 2.2156 | 17 | 7.9305 | 3.5824 | 23 |
| 1024 | 5.3636 | 2.4559 | 19 | 8.6614 | 3.7768 | 26 |
| 2048 | 6.2451 | 2.7046 | 22 | 9.7328 | 3.9570 | 29 |
| 4096 | 6.8345 | 2.8593 | 23 | 10.7832 | 4.3378 | 33 |
| 8192 | 7.9644 | 3.1142 | 26 | 11.6726 | 4.5118 | 35 |
| 16384 | 8.8550 | 3.3012 | 24 | 12.7799 | 4.7122 | 37 |
| 32768 | 9.8028 | 3.4937 | 25 | 13.8572 | 5.0067 | 38 |
| 65536 | 10.6125 | 3.5987 | 28 | 14.7832 | 5.1310 | 40 |

**Table 1. Hop count meaasurements**

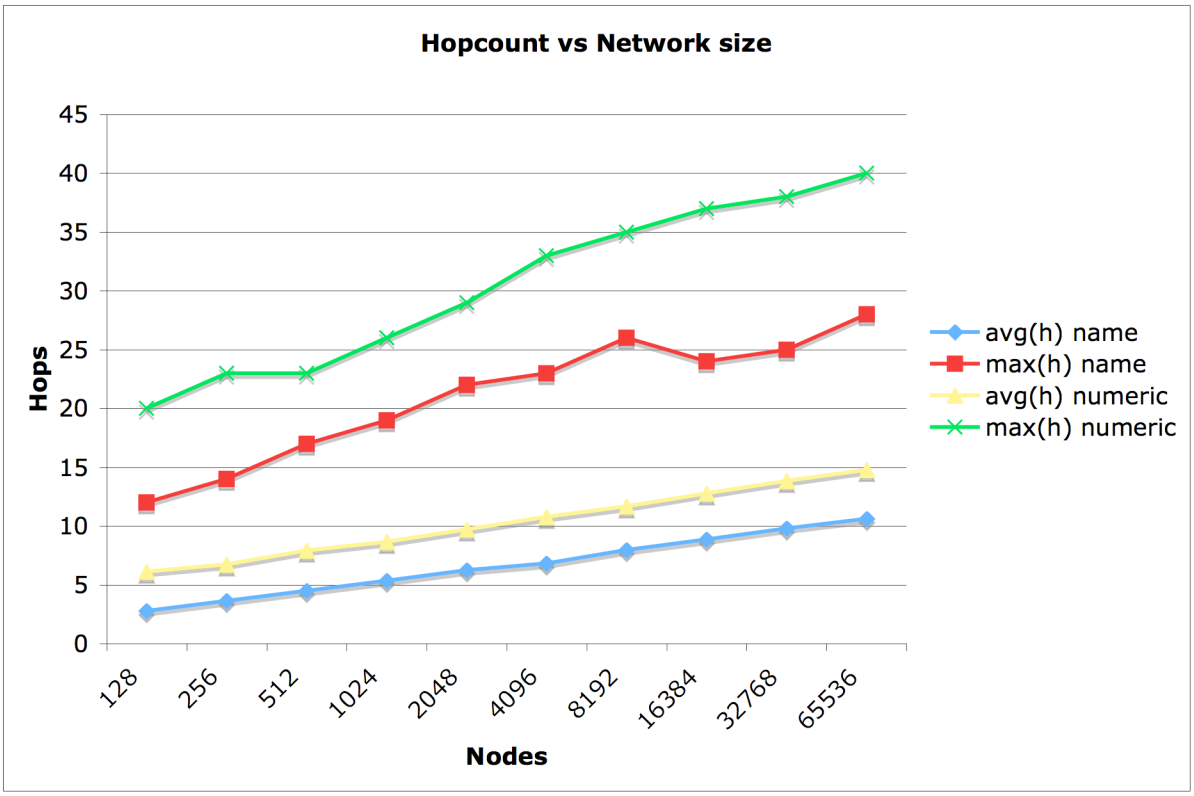


**Figure 2. Routing efficiency related to network size.**

## Number of Clusters

The goal of this experiment is to measure the number of isolated clusters formed after the random removal of large percentages of network nodes in a static skipnet. This experiment was conducted this way due to the lack of a detailed specification of the repairing process. For each network size, a percentage of nodes going from 50% to 99% is removed. The observed trends are shown in Figure **.** When a repairing process is active, results improve significantly.

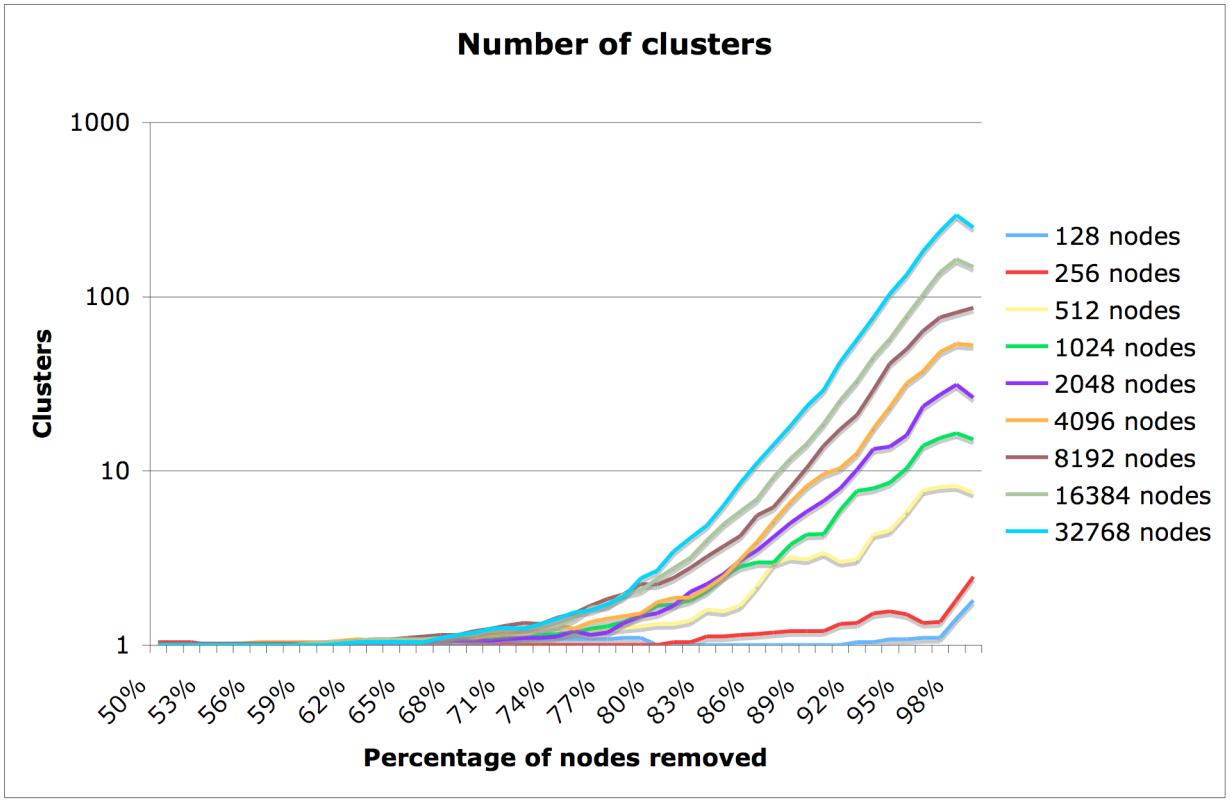


**Figure 3. Number of cluster after random removal.**

## Conclusions and future work

The goal of this project was to implement the Skipnet protocol using the Peersim Simulator. The basic protocol mechanism for DHT construction and routing infrastructure were implemented. The experimentation with Peersim allowed the verification of the small-world properties outlined in the original skipnet paper. In the future, it would be interesting to compare the performance of Skipnet with other DHT protocols using Peersim. An implementation using a Proximity Table could improve the results presented here. As for Peersim, I would like to see a transport layer that could come close to a connection oriented protocol, that would make a few life simpler on a few cases, but highly increases the demands of memory and processing power.