Paper Review: Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems

Sohail Ahmed Shaikh

January 21, 2019

1 Summary:

The motive behind this paper is to describe a decentralized distributed system aimed at facilitating object location and routing of messages in an overlay network of nodes to support peer to peer applications. The basic model of the system consists of assigining a unique nodeID to each node randomly. The authors then describe how routing of messages with a certain key to the nodeID closest to the key takes place. The routing of messages and the internal state (routing table) of the nodes are described in cases of faults, node removals and node additions to demonstrate the decentralized, scalable and self-organizing properties of the protocol. A fundamental feature of this system is it's good network locality properties.

2 Description:

A peer to peer application environment has properties such as decentralized control and self organization as nodes could be randomly added or removed from the network. Moreover such a network should be scalable and all nodes must have the same responsibilities and capabilities.
With these goals in mind the authors propose Pastry which acts as a generic substrate on top of which large scale peer to peer applications can be built.

The system has a simple design in which each node routes client requests and is assigned a 128 bit nodeID. This nodeID is randomly assigned when a node joins the system. The random assignment of the nodeID has an interesting result because this results in adjacent nodeIDs of having a high probability of being diverse in geography, ownership and jurisdiction.
Also, in a network of N nodes, for a given message and the concomitant key, Pastry routes it to the numerically closest node (having nodeID closest to the key) in at most $log_{2^b}N$ steps. Here b is a tunable parameter and will determine the number of entries in the routing table of each node. Pastry follows a prefix based routing to achieve this.

In brief, the routing of a message in Pastry works as follows: All the nodes in this scheme maintain a routing table, a neighborhood set and a leaf set. Each entry in the routing table has nodeIDs (with it's IP address) that share a prefix with the node on which this table resides.
The neighborhood set is a set of M nodeIDs and IP addresses that are the closest in distance (or some other proximity metric like IP routing hops) to the current node.
The leaf set is a set of L/2 nodes having numerically larger nodeIDs to the current nodeID and L/2 numerically smaller nodeIDs along with their IP addresses.
Using this internal state, the routing is accomplished.
In each routing step the message is forwarded to a node whose nodeID shares with the key a prefix that is

at least one digit longer than the prefix that the key shares with the current node's nodeID (this can be determined from the routing table); otherwise, the message is forwarded to a node whose nodeID shares prefix with the key and is numerically closer to the key than the current node's nodeID.
The authors claim that this simple routing scheme always converges and number of routing steps has an upper bound of $log_{2^b}N$.

The authors describe the self organization and adaptation properties of the system with respect to addition of new nodes in the system and removal or failures of one or multiple nodes. It is essential to maintain the routing tables of each of the nodes to provide consistency and accuracy in routing. The approach used in case of concurrent arrival and departures is an optimistic one as contention is rare. The number of messages exchanged have a bound of $O(log_{2^b}N)$ and nodes affected in these scenarios are only a small subset (not all nodes have to update their routing table).

The authors further extrapolate the good locality properties of Pastry in terms of routing table the routes and with respect to a proximity metric which could be IP routing hops. However, the assumption made here is that the triangulation inequality holds true for the distances between the nodes.
The case of arbitrary node failures and system performance in network partitions are also discussed.

Finally the performance of Pastry is evaluated in a system of 100,000 nodes which show good routing performance, high scalability with respect to the size of the routing table and number of hops and network failures.

3 Strong points:

1) The system is completely decentralized.
2) The authors experimentally proved the scalability of the system with random failures in a system of 100,000 nodes.
3) The system also has eventual consistency and self organization when nodes fail or join the system.
4) The routing table although more complex has a tunable parameter b through which we can manage the trade-off between the size of the routing table and the number of hops required to reach the destination.
5) A good contribution of the paper I feel, is that it provides a good substrate on top of which other peer to peer applications can be built.

4 Weak points:

1) The routing table is complex to accommodate locality guarantees. This might create overhead for maintaining the routing tables and waste bandwidth if nodes are frequently added or removed from the system.
2) Eventual consistency guarantees may not be sufficient if network latency is important for an application.
3) Solution to network partition and arbitrary node failures cannot be well handled by this system. In modern distributed systems with unreliable nodes this becomes increasingly important. The IP multicast solution that the authors propose for this.
4) In the current internet there are triangle inequality violations (TIV) and many studies have been carried out in this, for example the work by Lumezanu, et. al in "Triangle Inequality Variations in the Internet". The claim of the authors in Pastry providing good locality is lost due to the assumptions not being true as TIV delays in the internet are significant.

5 Improvements:

1) The causality of random unique IDs leading to good locality must be revisited as this claim does not hold true in the current internet with substantial trianle inequality violations. Instead of random nodeIDs they could be created more intelligently to ensure uniform distribution and locality.
2) The routing table should be made less complex as it could be difficult for applications to track the changes in such a system.