Project -3

Pastry Protocol

Failure Model (Bonus) Report

1. Team Members –

Kanika Gupta, 96977046, kanikagupta@ufl.edu

Nikhil Chopra, 98973831, nikhilchopra60@ufl.edu

2. Command to run the program –
Go to – project3bonus/lib

escript project3bonus numNodes numRequests numFailures

3. Working of Failure Model –

The user will enter the number of failure nodes in the command line while running the program. If the user does not enter anything, then number of failure nodes will be taken as numNodes/100 by default. After all the nodes' distributed hash tables are built, the failures will be created. All other functions work normally.

4. Failure Handling –

The failure needs to be handled if the node which is dead, is in the LeafSet or RoutingTable of any other node. In that case, we will have to remove that dead node from LeafSet and RoutingTable. To remove all the dead nodes from the LeafSet of a live node and add all live nodes, we need to check the farthest node that lies in the same side of the node which is dead either smaller or larger side. In this way, we can rebuild a node's LeafSet with all the live nodes.

We have done something opposite to this in case of rebuilding the RoutingTable for a node. The node that is dead and is present in another node's RoutingTable will have to ask its closest neighbours about any other alive node that can replace it.

5. Impact of Failure Handling technique –

Initially with the increase in number of failures, keeping numNodes and numRequests constant, the average number of Hops increase, because of inability of some nodes to get a proper replacement for dead node. But after increasing the number of failures further,

average hops decrease due to failed nodes and thus total live nodes in the network decrease.

numNodes	**numRequests**	**numFailures**	**Average Hops**
100	10	0	2.27
100	10	10	8.14
100	10	20	8.445
100	10	30	8.938
100	10	40	4.93
100	10	50	2.918