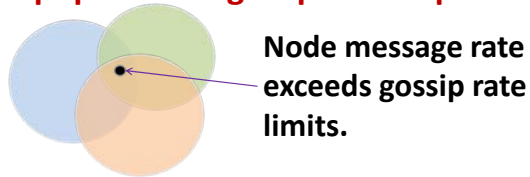


Problem

When gossip applications are popular and groups overlap...

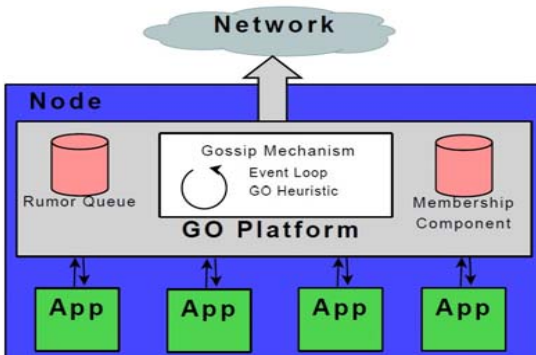


Wish list

Support multiple gossip groups with:

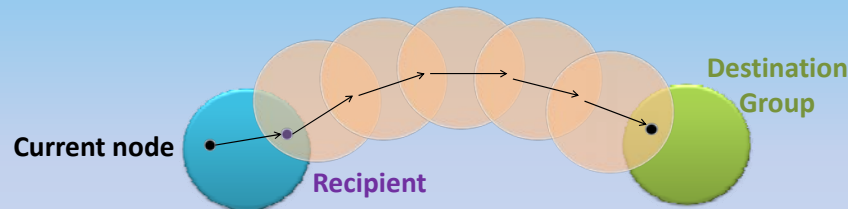
- Constant, low load.
- Standardized interface.
- Optimal target selection and gossip message contents.
- Group gets fair share of bandwidth.

Solution: Gossip Objects (GO)

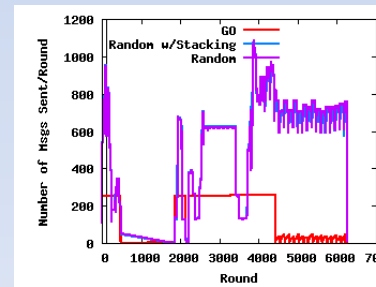
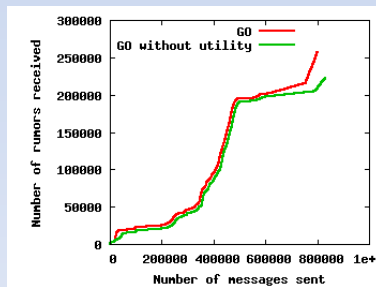


Approach and Results

- 1 Rumor stacking**
When possible, stack multiple rumors in a message. Data will reach the targets faster.
- 2 Traffic control**
Per-node limit of traffic independent of number of groups. Pick gossip partner based on group traffic rate.
- 3 Utility based indirect rumor delivery**
Include "indirect" rumors to increase delivery opportunities. Compute rumor **utility** to decide which rumors to include.



Utility: The proportion of uninfected nodes in the destination group by the time a rumor is expected to be delivered successfully.



Evaluation on a 55 minute trace of the communication layer of IBM WebSphere, with 127 nodes and 1364 groups.

Ymir Vigfusson



Ken Birman



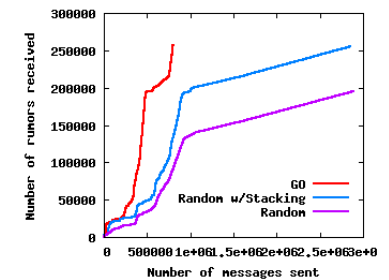
Qi Huang



Deepak Nataraj



GO is highly effective



Random dissemination gives the worst performance, but message stacking helps.

GO outperforms random stacking, delivering more rumors in less time yet sending fewer messages.