CLOUD COMPUTING CONCEPTS

with Indranil Gupta (Indy)

MULTICAST

Lecture D

RELIABLE MULTICAST

Reliable Multicast

- Reliable multicast loosely says that every process in the group receives all multicasts
  - Reliability is orthogonal to ordering
  - Can implement Reliable-FIFO, or Reliable-Causal, or Reliable-Total, or Reliable-Hybrid protocols
- What about process failures?
- Definition becomes vague

Reliable Multicast (under failures)

- Need all *correct* (i.e., non-faulty) processes to receive the same set of multicasts as all other correct processes
  - Faulty processes are unpredictable, so we won't worry about them

Implementing Reliable Multicast

- Let's assume we have reliable unicast (e.g., TCP) available to us
- First-cut: Sender process (of each multicast M) sequentially sends a reliable unicast message to all group recipients
- First-cut protocol does not satisfy reliability
  - If sender fails, some correct processes might receive multicast M, while other correct processes might not receive M

REALLY Implementing Reliable Multicast

- Trick: Have receivers help the sender

1. Sender process (of each multicast M) sequentially sends a reliable unicast message to all group recipients
2. When a receiver receives multicast M, it also sequentially sends M to all the group's processes

Analysis

- Not the most efficient multicast protocol, but reliable
- Proof is by contradiction
- Assumption two correct processes P*i* and P*j* are so that P*i* received a multicast M and P*j* did not receive that multicast M
  - Then P*i* would have sequentially sent the multicast M to all group members, including P*j*, and P*j* would have received M
  - A contradiction
  - Hence our initial assumption must be false
  - Hence protocol preserves reliability

Next

- Combining fault-tolerance and multicast