Recovery

- Techniques thus far allow us to tolerate faults
- Recovery: operations that must be performed after a failure to recover to a correct state
- Techniques:
  - Checkpointing
  - Message logging

Checkpointing

- Each process periodically checkpoints its state
- Upon a crash roll back to a recovery line, i.e., the most recent consistent collection of checkpoints.
Independent Checkpointing

- Each process periodically checkpoints independently of other processes.
- Upon a failure, work backwards to locate a recovery line.
- Problem: if most recent checkpoints form an inconsistent cut, will need to keep rolling back until a consistent cut is found.
- Cascaded rollbacks can lead to a domino effect.

Coordinated Checkpointing

- Take a distributed snapshot [discussed in Lecture 9].
- Upon a failure, roll back to the latest snapshot.
  - All processes restart from the latest snapshot.
Message Logging

• Checkpointing is expensive
  – Taking a snapshot is expensive
  – Infrequent snapshots => all computations after previous snapshot will need to be redone [wasteful]
• Combine checkpointing (expensive) with message logging (cheap)
  – Take infrequent checkpoints
  – Log all messages between checkpoints to local stable storage
    • Sender-based logging: log outgoing messages before sending them off
    • Receiver-based logging: log incoming messages before delivering them to the application
  – To recover: simply replay messages from previous checkpoint
    • Avoids recomputations from previous checkpoint