Transactions

Client/Server with Transactions: ACID

- Atomicity
  - All-or nothing
- Consistency
  - Leaves system in a correct state
- Isolation
  - Concurrency control
  - Serialized access to shared resources
  - Changes not visible until commit
- Durability
  - Effects are permanent after commit (written out to disk)
  - Persistent = durable
- Trick: everyone must play
The Flat Transaction: All-Or-Nothing

Resources are “tied up” for the duration of a flat transaction

Locks

Flat transactions should be small in scope

Surgical strike approach

Come in, do the damage, get out—no lingering!
Limitations of Flat Transaction

- Compound business transaction
- Travel reservation: air, car, hotel
- Partial rollback?
- Humans in the loop
- Long think time ties up resources
- Long-lived transactions
- CAD, document management, manufacturing
- Check-in/Check-out, versioning, workflow
- Bulky transactions
- Inter-business transactions

Flat Transactions: Commit or Start All Over
Flat Transactions and Think Time

Long-Lived Transactions
Flat Transactions and Bulk Update

Update 1,000,000 Records

Extranet B-to-B Transactions

Company A Transaction

Commit

Local DB

Commit

MOM Transaction

Commit

Company B Transaction

Commit