Reliability

Fault-tolerant for human errors

- Design systems in a way that minimizes opportunities for error
  - e.g., add constraints to interface to encourage the right things
- Decouple the places where people make the most mistakes from those where they can cause failures
  - e.g., try the change first in fully featured non-production sandbox environment
- Test thoroughly at all levels, from unit tests, integration tests, and manual tests
  - e.g., automation is quite important
- Allow quick and easy recovery from human errors
  - e.g., possible to roll back changes, roll out changes gradually, each to recompute data
- Set up detailed and clear monitoring (telemetry)
  - e.g., performance metrics, error rates can provide hints for troubleshooting and fixing
- Implement good management practices and training

Scalability
Describing Load

- Load parameters
  - the volume of reads or writes
  - the volume of data to store
  - the complexity of the data
  - the response time requirements
  - the access patterns
- Some examples
  - requests per second to a web server
  - the ratio of reads to writes in a database
  - the number of simultaneously active users in a chat room
  - the hit rate on a cache