1 Distributed Systems

Define distributed systems. Explain the different kinds of communicating entities and communication paradigms. Give an overview of different distributed system architectures.

Distributed system = a system of ~~components~~ subsystems communicating over networks by sending messages
Scalability = How is performance impacted when the number of users grow?
Failure handling = When something fails, find a way of accomplishing your goal
Transparency = The user doesn't care about the distribution. Neither does the developer.

Entities
1. Objects: defined in interfaces
2. Components: address the weaknesses of objects, but all dependencies are explicit => self contained units
3. Web services: defined by URI, with well-defined behavior; interacts using XML-based message exchanges via internet protocols

Communication paradigms
1. Interprocess: low-level, direct access to the API via sockets, support for multicast communication
2. Remote invocation: calling a remote operation, procedure or method
    a. Request-reply: primitive, supports the other two, used in http
    b. Remote procedure call: calling procedures on remote computers as if they were local; abstracts distribution/encoding
    c. RMI: resembles RPC, but with distributed objects: a calling method can invoke a method on a remote object
3. Indirect: through an intermediary , with no direct coupling
    a. Group comm
    b. Publish-subscribe
    c. Message queue
    d. Tuple spaces: common shared persistence
    e. Distributed shared memory

Coupling
- Time
- Space

Architectural model
- Layering: abstract operations between different layers
    ○ Apps and services
    ○ Middleware: provides a high-er level abstraction
    ○ OS
    ○ Hardware & network
- Tiers: complementary to layers; 2/3 tiers
- Thin clients
- Proxies
- Brokers: between requester and service provider
- Reflection