

Service-oriented architecture (SOA)

- Software models we have discussed so far are “monolithic” designs: a single system with many components, designed to meet a specific set of needs
- Service-based designs take very different approach: build many independent components, each designed to do one specific task
- To perform a more complex task, specify which services to connect/use in what sequence
- Each service is, in essence, a constantly re-used component

Microservices

- Similar idea to SOA, but making the each service (microservice) focus on one really small task
- Cloud-driven approach, have many instances of each service running simultaneously
- When a task is running, it looks for an available instance of whichever microservice it needs next
- Resistant to failures in individual networks/devices: if one instance of a microservice crashes/fails, grab another

Microservices cont.

- Each microservice should be able to run independently of other microservices
- Each microservice type will have unique identifier and a specific protocol you must follow to interact with it
- Like concept of function profile, but adds the element of sequencing – what must be sent and what will be returned, in what order