Midterm Review

CSCE 742 - Lecture 12 - 10/16/2018

The Midterm

- 75 minutes, in-class
- Closed book, no notes!
- Graded on quality of answers, not how much you wrote.
- Very similar in format to the practice midterm
- Study your homework, in-class activities.

The Midterm

- Topics:
  - Introduction
  - Viewpoints and Perspectives
  - Architectural Definition Process
  - Context
  - Stakeholders
  - Scenarios
  - Concerns and Principles
  - Functional View
  - REST Style

General Questions

- Today: Go over practice midterm questions.
- First - any general questions on course content or homework?

Question 1

How is Software Architecture different from Software Design? How is it the same?

Question 1

How is Software Architecture different from Software Design? How is it the same?

- Architecture is the partitioning of a system into smaller chunks that can be developed independently and integrate to create a functioning system.
  - Architectural decisions (from a functional view) are a subset of the design decisions: those that affect this partitioning.
  - Architecture focuses on non-functional qualities that are not traditionally associated with software design.
- Design is a continuum. It can be difficult to assess whether a design decision is architectural or not, and this is also a matter of perspective.
  - A “design detail” for an architect may be a critical architectural decision for a designer.
  - Design tends to be low level, how we realize the requirements. Architecture bridges abstract requirements to detailed design.

Question 2

1. What are differences between requirements and goals?
2. List two requirements and two goals for the airport parking example.

Question 2

Differences between requirements and goals

- Requirements are specific and testable
- Goals are “fuzzier” and harder to quantify in terms of the expected system behavior.
  - Goals are higher level than requirements and a single goal can be refined into several requirements.
- Goals are still important.
  - Consistent theme and presentation to the user can be as important as the functional behavior of the device.

Question 2

List two requirements and two goals for the airport parking example.

Requirements

- Under normal operation, the system shall raise the gate within one second of user confirmation of payment.
- During a loss-of-communication event with the credit card service, the system shall record credit transactions to deferred transaction log for later processing.

Goals

- The airport parking system should be usable to people without any experience with computer systems.
- The airport parking system should be assembled, when possible, of COTS components to minimize cost and maintenance effort.

Question 3

Viewpoints allow separation of concerns and prevent overwhelming users with information. In many cases, viewpoints are tightly coupled (a change in one viewpoint necessitates a change in another viewpoint).

Consider the Deployment and Concurrency viewpoints.

1. Give examples of information that is duplicated between the two viewpoints and unique to each viewpoint.
2. Should these two viewpoints be merged? Argue in terms of cohesion and coupling of the information between the two views.

Question 3

Give examples of information that is duplicated between the two viewpoints and unique to each viewpoint.

- Processes and interconnections are duplicated between viewpoints.
- Physical resource containing the process is unique to the deployment view.
- Synchronization primitives are unique to the concurrency view.

Question 3

Should these two viewpoints be merged? Argue in terms of cohesion and coupling of the information between the two views.

- You duplicate portions of system across viewpoints.
- Could overlay physical resources on the concurrency model or synchronization primitives on deployement model.
- However, reasons for constructing each model differ.
  - Deployment is concerned with physical resources. Concerns of operations and support staff.
  - Concurrency model is concerned with logical issues such as race conditions and deadlock, concerns of development and testing team.

(You can argue either way, but make a good case)

Question 4

Are the following business principles, technology principles for a specific system, or low-level object-oriented design advice? Explain why you chose each answer.

1. All external access to the system should be via two factor authentication, involving something you have and something you know, e.g. smart card and password
2. Security measures should be applied appropriately to the level of risk defined for a given system or database.
3. Encapsulate actions into objects that can be stored, replayed, or undone.
4. Adopt an appropriate level locking strategy, typically using optimistic locking for frequently changed data and pessimistic for data where there is a low rate of change.
5. Minimize the number of security interactions for the web store.

Question 4

1. All external access to the system should be via two factor authentication, involving something you have and something you know, e.g. smart card and password
   Technology principle. Two-level authentication would not be required for all data across an enterprise.

2. Security measures should be applied appropriately to the level of risk defined for a given system or database.
   Business principle. Broader than a single application.

3. Encapsulate actions into objects that can be stored, replayed, or undone.
   Low-level design advice. Part of command pattern. Discusses objects. Does not rise to level of architectural principles.

Question 4

4. Adopt an appropriate level locking strategy, typically using optimistic locking for frequently changed data and pessimistic for data where there is a low rate of change.

   Technology principle or design advice (depends on argument). Would argue tech principle - describes a policy on relational databases, irrelevant to many applications. Could be considered a policy over all applications using a database.

5. Minimize the number of security interactions for the web store.

   Business or technology principle (depends on argument). If "web store" is one application, this is a technology principle. If a network of connected services, this is a business level principle.

Question 5

Making servers stateless (in terms of session state), like in the REST style, has many benefits for scalable and reliable client/server systems.

Describe at least three drawbacks of building stateless systems.

Give two examples of systems where constructing stateful servers is a good idea.

Question 5

Describe at least three drawbacks of building stateless systems.

- More complex programming model.
- Can make application much slower to reconstruct state.
- May require significant rewrite of existing systems (expense).
- May require substantially more server resources to host due to state reconstruction.

Question 5

Give two examples of systems where constructing stateful servers is a good idea.

- Applications where initialization and caching requires significant amount of state
  - (medical imaging, CAD/CAM, many others)
- Applications where it would be cost-prohibitive to rewrite
  - Legacy applications written using service-oriented architectures.

Question 6

In class and in the Garlan and Shaw paper, we considered architectural styles, including: **pipe and filter**, **event-based**, **layered**, **repository**.
You are designing an automotive system. For each style, describe a subsystem where the style would be an appropriate structuring mechanism - and why - or describe why this style does not apply to any of the subsystems.

a. On-star communications: manages communications with satellite
b. sensor management: turns noisy sensor data into useful information
c. motion control: operates the motors and provides position and velocity
d. Image processing system to identify highway lanes
e. UX vehicle management involving touch screen
f. Health/status monitoring: checks status of all other subsystems to ensure correct operation
g. Collision avoidance system
h. Dashboard displays

Question 6

Pipe-and-Filter:

- Data is transformed by one process and passed as input to the next.
- Sensor management is often handled using this style.
  - Data de-noising, sensor fusion transformation take sensor data, transform it, and pass it forward.
- Communication
  - Filters for compression and encryption could use this style.

Question 6

Event-Based:

- Control model. Events trigger actions in an affected subsystem.
- Supervisory control is event-based.
  - When objective reached, send an event to replan.

Question 6

Layered:

- System is architected as a series of layers, only communicating with layers directly above and below.
- Natural layering between supervisory control, navigation control, and motion control.
- Communications systems are often layered (TCP/IP/Physical communication layer)

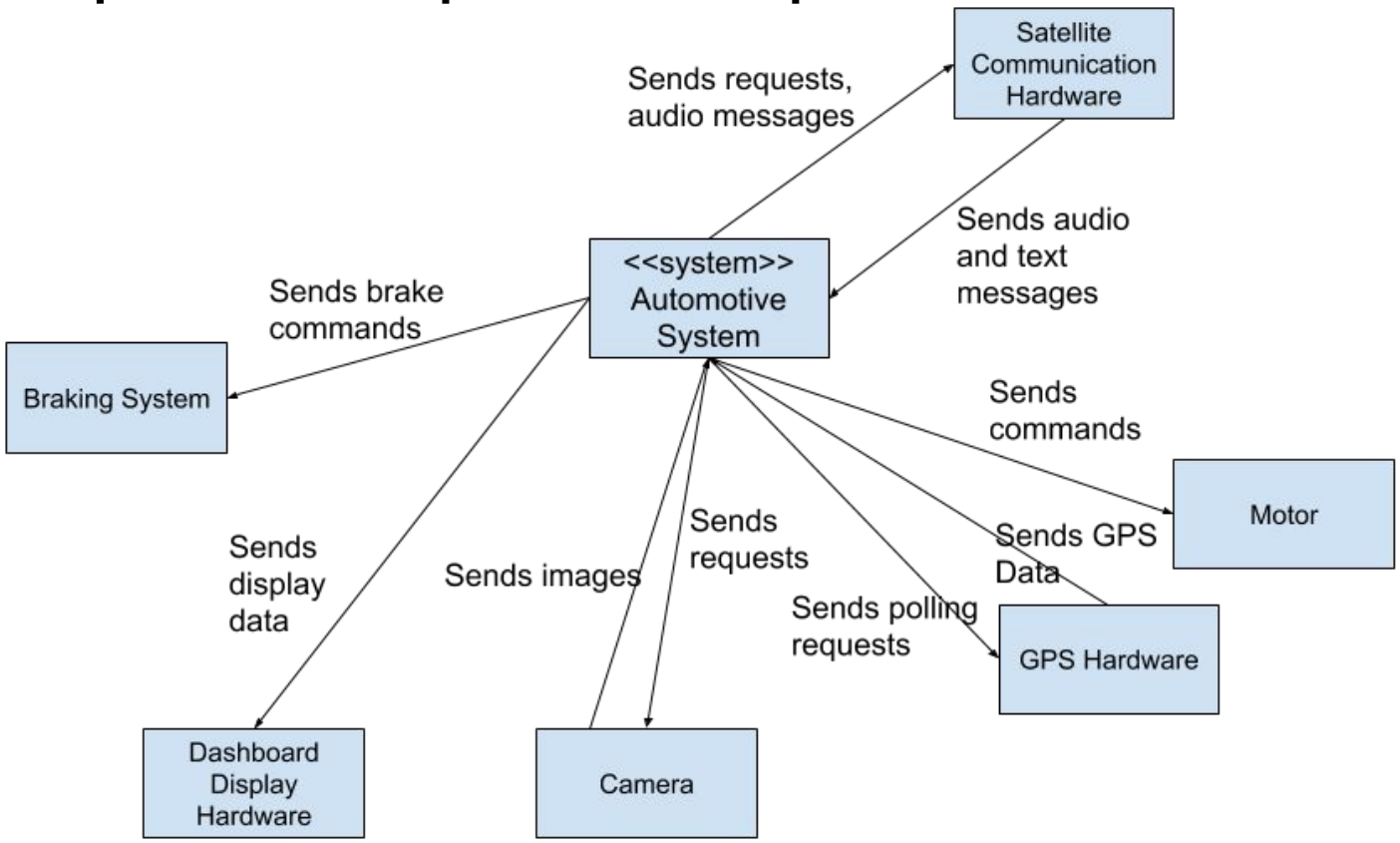
Question 6

Repository:

- Subsystems all work with a large central data repository.
- Health/Status Monitoring uses this style.
  - Vehicle health from many systems aggregated in one place.
  - Central repository of sensor data could underlie other systems as well.

Question 7

Create a context diagram for the automotive example in the previous question.



Question 8

Consider the software for air-traffic control at an airport (say, CAE).

Identify one performance, one availability, and one usability requirement that you think would be necessary for this software and develop a quality attribute scenario for each.

Question 8

Performance Requirement: Under normal load (< 500 aircraft), displayed aircraft positions shall be updated at least every 50 ms.

Performance Scenario:

- Overview: Check system responsiveness for displaying aircraft positions
- System and environment state: Deployment environment working correctly with less than 500 tracked aircraft.
- External stimulus: 50 Hz dispatch of ATC system.
- System response: radar sensor values are computed, new position is displayed to the air traffic controller with maximum error of 5 meters.
- Response measure: Display process completes in less than 50 ms.

Question 8

Availability Requirement: The system shall be able to tolerate the failure of any single server host, graphics card, display or network link.

Availability Scenario:

- Overview: One of the monitor display cards fails during transmission of a screen refresh.
- System and environment state: System is working correctly under normal load with no failures.
- External stimulus: display card fails
- Required system response: display window manager system will detect failure within 10 ms and route display information through redundant graphics card with no user-discernable change to ATC aircraft display. Graphics card failure will be displayed as error message at bottom right hand of ATC display.
- Response measure: no loss in continuity of visual display and failover with visual warning completes within 1 s.

Question 8

Usability Requirement: The user interface shall be designed to minimize user mistakes by providing a route projection and requiring confirmation of decisions. Users shall make fewer than two mistakes per month.

Usability Scenario:

- Overview: A user is tasked with making a routing decision.
- System and environment state: System is working correctly under normal load with no failures.
- External stimulus: A pilot indicates readiness to take off, and the user selects an initial trajectory.
- Required system response: Display window manager shall show a projection for that trajectory, and ask the user to confirm their decision.
- Response measure: The user shall select and confirm a route that could lead to a potential collision fewer than two times per month.

Question 8

Security Requirement: The system shall maintain audit logs of any logins to the ATC database.

Security Scenario:

- Overview: a malicious agent gains access to flight records database.
- System and environment state: System is working under normal load.
- External stimulus: malicious agent obtains access to the flight records database through password cracking, and downloads flight plans for commercial aircraft.
- Required system response: audit log contains login and download information to support future prosecution of user.
- Response measure: system audit contains time, IP address, and related information w.r.t. download.

Any additional questions?

Next Time

- Midterm!
  - Practice midterm, answers, these slides on site.

- Homework:
  - Project Part 2 - Due tonight
  - Assignment 2 - Due on the 25th