Repetition Questions (dt. Wiederholungsfragen) Lesson 1

Link to Lecture (via Fact Sheets)

The first lesson of the lecture today covered the following concepts (see fact sheets in script folder):

1. Software Architecture Fundamentals: Definitions, Phases, Role of the Architect[1]
2. Architecturally Significant Requirements (ASRs)[2]
3. Views and Viewpoint Models[3]

In the corresponding exercise[4], we worked with ASRs and Kruchten's 4+1 views in the context of a particular project. In the self study assignment that will be handed out next week, you will have the opportunity to practice the concepts one more time by yourself.

Topic/Concept: Fundamentals (Definitions, Role, Phases)

1. How many definitions of software architecture were given in the lecture, and what are the leading concepts in each of them?
2. List at least three software-intensive systems that are publicly visible or were featured in the lecture, and enumerate two to three of the architectural elements in each of these systems.
3. Name the three architecting phases and list three relations between them.
4. List at least three external stakeholder interactions an application architect might have during a typical business day (in the form "[interact] with [role] to [purpose]", e.g. "meet with domain analyst to discuss number of domain model elements that have to be stored persistently").

Topic/Concept: Architectural Significance

1. What is the motivation for the concept of architectural significance?
2. How can the architectural significance of a requirement, element and/or decision be assessed?
3. Does one particular requirement always have the same significance, independent of the current project and its context?
4. What do you do if many or all requirements turn out to be highly significant architecturally?

Topic/Concept: Viewpoints

1. What is the benefit and purpose of a viewpoint concept?
2. Which viewpoint models were listed in the lecture slides and in the fact sheet?
3. What is the difference between the process viewpoint and the physical/deployment viewpoint in Kruchten's 4+1 model (hint: stakeholder concerns, notation)?
4. How do viewpoints and architectural significance/ASRs relate to each other?

[1] ../2-lecture-script/lesson1/ZIO-ArchitectureFundamentalsFactSheet.pdf
[2] ../2-lecture-script/lesson1/ZIO-ArchitecturalSignificanceFactSheet.pdf
[3] ../2-lecture-script/lesson1/ZIO-ViewpointFactSheet.pdf
[4] ../3-exercises-solutions/ZIO-AppArch-ExerciseWeek1.pdf

Answers

Topic/Concept: Fundamentals (Definitions, Role, Phases)

1. *Three definitions, one emphasizing structure (i.e., components and connectors) and one centered on decisions; finally, a hybrid one talking about structure and principles.*
2. *Core banking SOA, order management SOA, Distributed Control System (DCS); see slides 4, 5, 6 in lecture for top-level components and relationships (e.g., "business function" and "BS1"). In the exercise, we come across Spinnaker components such as Orca and external dependencies such as Kubernetes (without going into detail).*
3. *Architectural Analysis yields the ASRs that drive Architectural Synthesis; in Architectural Evaluation, the design that comes out of Architectural Synthesis is reviewed w.r.t. coverage and satisfaction of the ASRs from Architectural Analysis.*
4. *Examples: a) negotiate effort with project manager and project sponsor, b) coach developer to apply an architeturally evident coding style, c) elicit NFRs with business person (see slide 9 in lecture) and reading recommendations in fact sheet*

Topic/Concept: Architectural Significance

1. *The notion of arch. significance and ASRs helps to scope work and to stay focused and make sure one works on relevant problems. To quote a lean principle, it makes sure to "avoid waste" and to meet the NFRs for the system under construction eventually.*
2. *Requirement criteria (W-x). element checklist (E-y), decision questions (T-z)*
3. *No, context matters, consultants like to give the "it depends answer" (and it is not always an excuse or a request for funding)*
4. *This actually happens all the time in practice. Some strategies and tactics are: perform additional risk- and cost-based prioritization; challenge and (re-)negotiate requirements; ask for additional resources*[5]

Topic/Concept: Viewpoints

1. *Complexity management and stakeholder focus: make sure not to forget important stakeholders and their concerns, and to communicate only relevant information to them*
2. *4+1 model by P. Kruchten used in (R)UP, IBM viewpoint model (in fact sheet), Rozanski/Woods (in fact sheet)*
3. *System integrator vs. system administrator; more dynamic vs. more static (in the original model); interoperability and concurrency concerns vs. performance and scalability*
4. *Many ASRs are cross-cutting concerns that become perspectives in two-dimensional viewpoint models, for instance performance and security.*

[5] "Leave "project" should not be one's first reaction, but actually is required occasionally, when chances of succeeding are too low (how to time and execute such exit from a mission impossible without damaging professional relationships and project track record makes an interesting discussion topic, for another time).