Exam Software Architecture - Answers

6 November 2017, 17:00 – 20:00

This closed-book exam consists of 4 questions on 6 pages. The 7th page can be used for answering question II.A. Please check first whether you have properly obtained **all** pages.

Some tips:

- Answer on the separate exam papers.
- Disengage the last page for answering question II.A.
- Read carefully the question before you start answering!
- Be clear and concise in your answers
- Reread your answers to check whether you really answered the question posed!
- Write clearly!
- Once finished, fold all papers together and hand in your work.

Good luck with the exam!

Question	Max. points	Awarded points
1	25	
2	25	
3	30	
4	20	
Total:	100	
Grade:	Total / 10	

This exam contains answers to the questions.

Note that for many questions, this is not THE answer, but a sufficient answer to gain all points.

I. Basic concepts (25 points)

We consider software architecture to be "the set of structures needed to reason about the system, which comprises software elements, relations among them, and properties of both" (Bass et al, 2012),

A. One can argue that an architecture is actually a structure of structures needed to reason about the system. Explain why (5p).

A structure is a set of elements, together with relations on those elements (1p). In an architecture, each of the views and structures together should be coherent and consistent (1p). Especially the relations between the different structures are important, e.g. traceability (1p); hence an architecture can be seen as a set of structures with relations between them (1p), and thus is a structure (1p).

B. Not having an inconsistent architecture documentation is a key task of the architect. Inter-views, defining the relations between the different architectural elements in different views assists in this. Explain whether these inter-views are a viewpoint or a perspective on the architecture (5p).

Inter-views define relations between the different views: traceability (2p). Hence, they can be seen as a structure overarching the different views, and thus a perspective (3p). If an inter-view is seen as a view, to which viewpoint should it belong?

In essence, software realization can be seen as creating a mapping from the architectural artifacts to the realized software units.

C. Explain whether this mapping is an architectural view, and/or part of the realized software units (5p).

Both (3p): Architecture view: mapping from architectural elements to software units (1p). Software view: annotations in the source code to which elements they belong (1p)

One of them? Max 3p

D. Explain why Architecture Compliance Checking and Architecture Reconstruction can be seen as two sides of the same coin (5p).

Architecture compliance checking is the activity of validating whether a mapping from software units to an architecture is consistent (1p). In compliance checking, the architecture is given as a hypothesis (1p), in architecture reconstruction one tries to find a hypothesis (1p), and then can use compliance checking to validate the reconstruction (2p).

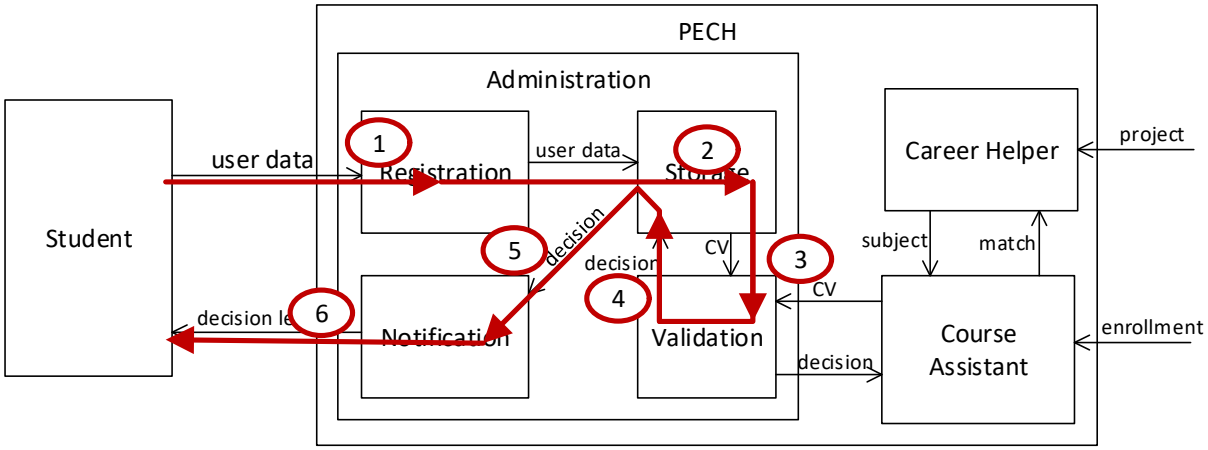
E. Explain the relations between a model, a view and a structure (5p).

A model is an abstract representation with notation, syntax and semantics to describe some system (1p). Therefore, it can be used to describe a structure (2p).
To document the structure, the architect uses views. (2p).

Only the definitions? Max 2p

II. Views and their consistency (25 points)

The Private Education and Career Helper (PECH) is a system that helps students in finding the best courses and career prospects. One of the views is the Functional Architecture Model as depicted below.



As shown in the FAM, the Administration module consists of four main submodules: Registration, Validation, Storage and Notification. After Registration, the data is sent to the Storage submodule. In 10% of the cases, the Storage submodule passes the user data to the Validation module. In all other cases, the Notification submodule sends a message to the user. Unfortunately, in 1 out of the 5 times, Notification fails. In that case, the user data is passed back into the Storage submodule. In total 70% of the output of the Validation submodule is fed back into the Storage submodule.

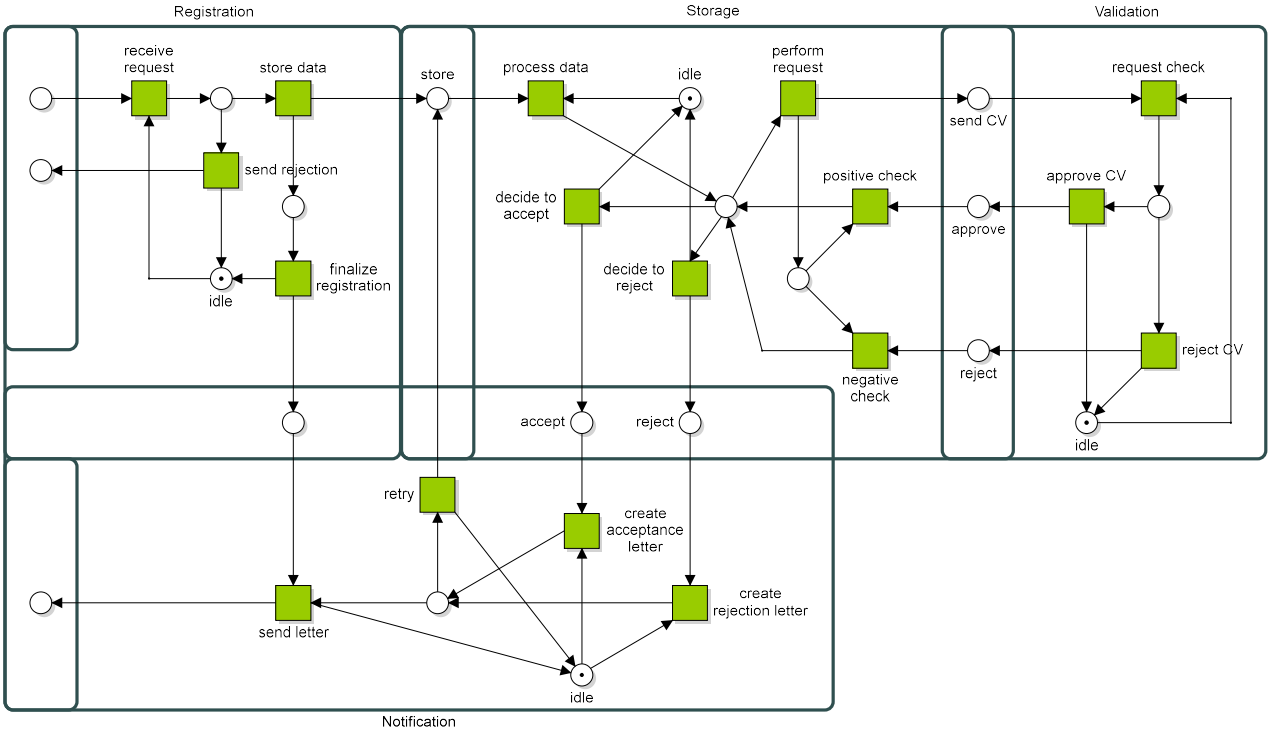
A. Draw a scenario that best describes the above scenario as an overlay on the FAM given on the answer sheet (5p).

Scenario without validation module: max 4p (rationale: the scenario with validation covers all aspects described in the text)

Scenario with also course assistant: -1p

No numbers or other indication of the flow order: -3p

One of the architects want to analyze the protocols between the submodules of the Administration module, and delivers the following Petri net.



B. Explain to which viewpoint this Petri net belongs (5p).

Concurrency (2p) It depicts the internal behavior and communication of the different modules, including concurrent behavior, in the system (3p), rather than the information flow in the system

C. An important property of architecture documentation is consistency. Are (1) the Petri net, (2) the scenario, and (3) the FAM consistent with each other? Explain your answer (2+8p).

Both (1) and (2), and (1) and (3) are not consistent: Registration sends a message to notification, while this arrow is not present in the FAM nor the scenario. (4p + 4p)

(2) and (3) are consistent, as each step in the scenario is a flow in the FAM. (2p) NB: If the scenario in A considers a back-and-forth between Notification and Storage, it is inconsistent (2p)

An important notion in modelling is correctness: the model should be correct, and it should be the correct model. For Petri nets, weak termination and proper termination are two important concepts that together form the correctness notion considered in the lectures.

D. Is the above Petri net correct? Explain your answer (5p).

No. for correctness, we need to consider the net without the open interface places (the skeleton). In that case, Registration is a generator, and thus the place store is unbounded. Thus after firing the transitions "receive request", "store data" and "finalize registration", all components are in their final state, but store contains a token. Hence, the net is not properly terminating. (5p)

III. Quality properties and their analysis (30 points)

An architect not only considers functional requirements, but also quality properties.

A. Explain how the viewpoint catalogue can assist the architect in analyzing a quality property (5p).

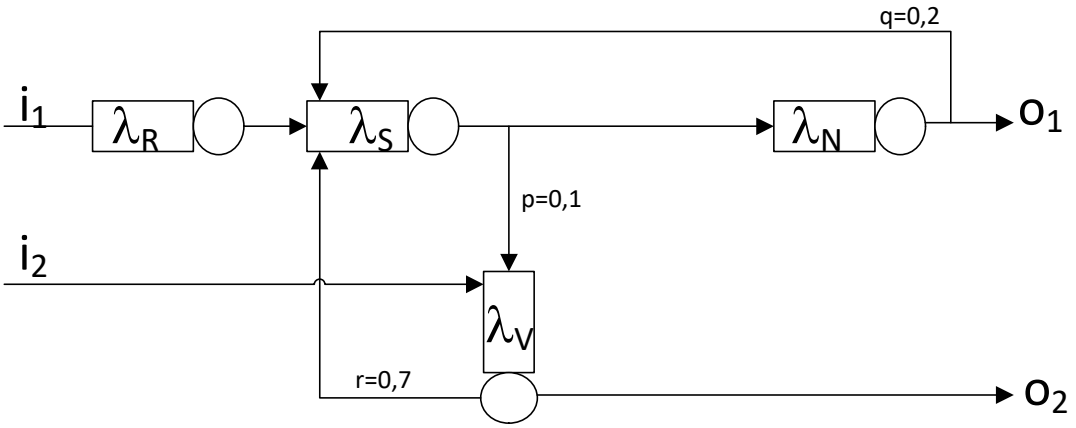
Quality properties address many different views. The catalogue categorizes these views (1p) and gives guidelines, modelling techniques and standard practices for each set of views that address similar concerns. (1p) In this way, analysis of quality properties can be directed, and upfront defined using the viewpoints (1p). In this way, the architect can standardize the analysis techniques, use checklists and ensure (s)he did not forget to address any relevant view for that QP. (2p)

B. One way to address a quality property is through quality scenarios. Such a quality scenario can be seen as an architectural view. Explain to which viewpoint this view belongs best. (5p).

Context (2p). It addresses the system (or part of it) as a black box by stating what the response should be and how this can be measured from the outside on certain stimuli (3p)

Consider PECH as introduced in Question II. The Administration module is critical for the performance of this system. Therefore, architect decides to do an additional analysis. As shown in the FAM, the Administration module has two incoming sources: one into Registration, and one into Validation. The former receives 4 requests per minute, the latter receives 5 requests per minute. Registration is expected to take 10 seconds to handle a single request. The Storage submodule has a service time of 15 requests per minute. The Notification and Validation submodules both handle 10 requests per minute.

C. Draw the queueing network ("Jackson") that describes the above system (5p).



Per error: -1p

D. Calculate the output rate of the Validation submodule to the Course Assistant module, following the FAM in Question II and the queuing network drawn above for question III.C (10p).

First set up the arrival rates (4p):

- $\lambda_R = i_1$
- $\lambda_S = \lambda_R + r \cdot \lambda_V + q \cdot \lambda_N$
- $\lambda_V = i_2 + p \cdot \lambda_S$
- $\lambda_N = (1-p)\lambda_S$

Remark that the outrate asked is $(1-q)\cdot\lambda_V$ (2p)

Actual calculation (3p)

$$\lambda_S = \lambda_R + r\cdot\lambda_V + q\cdot\lambda_N = i_i + r(i_2 + p\cdot\lambda_S) + q(1-p)\lambda_S$$

$$\equiv \lambda_S - r\cdot p\cdot\lambda_S - q(1-p)\lambda_S = i_1 + r\cdot i_2$$

$$\equiv \lambda_S = \frac{i_1 + r\cdot i_2}{(1 - r\cdot p - q(1-p)}$$

Filling in gives: $\lambda_S = \frac{4+0.7\cdot 5}{(1-0.7\cdot 0{,}1-0{,}2(1-0{,}1)} = \frac{7.5}{1-0{,}07-0{,}18} = \frac{7.5}{0.75} = 10$

Calculation of the outrate (1p):

$$(1-r)\cdot\lambda_V = (1-0{,}7)\cdot(i_2 + p\cdot\lambda_S) = (1-0{,}7)(5 + 0{,}1\cdot 10) = 0{,}3\cdot(5+1) = 0{,}3\cdot 6 = 1{,}8$$

E. Calculate the expected total time items spent in the Administration module (5p).

See that it is the sum of the different servers (1p)

See that the Sojourn formula should be used (1p)

Writing down the sojourn formulae and fill in the right values (2p)

- $\mu_R = 10\ s = 6\ requests/min$
- $\mu_S = 15\ requests/min$
- $\mu_V = 10\ requests/min$
- $\mu_N = 10\ requests/min$
- $\lambda_R = 4$
- $\lambda_S = 10$ (Result from III.D)
- $\lambda_V = 6$ (Result from III.D)
- $\lambda_N = (1-p)\lambda_S = 0.9\cdot 10 = 9$

$$\frac{1}{\mu_R - \lambda_R} + \frac{1}{\mu_S - \lambda_S} + \frac{1}{\mu_V - \lambda_V} + \frac{1}{\mu_N - \lambda_N} = \frac{1}{6-4} + \frac{1}{15-10} + \frac{1}{10-6} + \frac{1}{10-9}$$

Actual calculation (1p)

$\frac{1}{2} + \frac{1}{5} + \frac{1}{4} + 1 = \frac{10}{20} + \frac{4}{20} + \frac{5}{20} + 1 = \frac{10+4+}{20} + 1 = \frac{19}{20} + 1 = 1.95$

M/M/1 formulae

- Utilisation: $\frac{\lambda}{\mu}$
- Expected number of elements in a node: $\frac{\rho}{1-\rho}$
- Expected number of elements in the queue: $\frac{\rho^2}{1-\rho}$
- Expected waiting time: $\frac{\rho}{\mu-\lambda}$
- Expected sojourn time: $\frac{1}{\mu-\lambda}$

IV. Architectural styles and product lines (20 points)

Given is the definition of a Product Line: A product line of software is a set of software-intensive systems sharing a common, managed set of features that satisfy the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way.

A. Explain the difference between a variation point and a variation mechanism (5p).

A variation point is a location in the architecture where variability is realized, so that different sets of features can be used (1P). A variation mechanism is a software construct that provides a generic interface to the different behavioral or quality property characteristics (1P). Hence, the difference is that the point states where and what can be varied (2P), whereas the mechanism states how this is realized (1P).

Consider PECH again. Notifications are delivered by sending physical letters. PECH now wants to support multiple ways of sending the notification, including email.

B. Explain how quality scenarios can be used to define a variability point for the above situation at PECH, **and** give an example of such quality scenario for PECH (3+2p).

A quality scenario can be used to define which actor requires which variation, and how the system needs to be adapted to support that variation: the source defines the requested variant, the stimulus is the request, the response is which variants will be supported, and the measure defines the cost/time to create the core assets and the variants. (3P)

Environment: Running deployment of PECH

Source: Actor requests notification by Email, SMS and WhatsApp.

Stimulus: Request for notification by different communication means

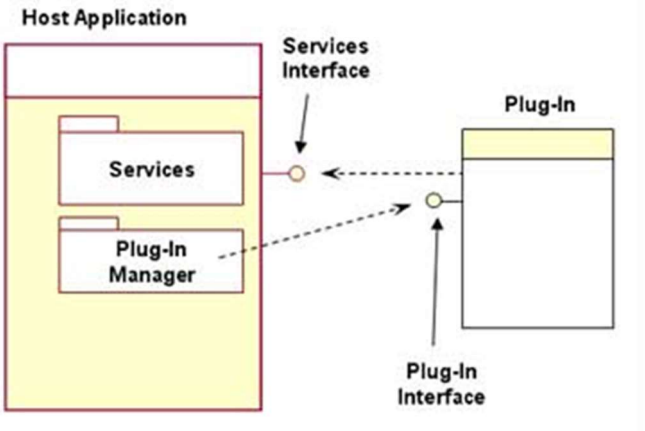
Artifact: PECH

Response: Selected communication means (e.g. email, SMS, WA)

Measure: generic notification platform , user notification specification, specific notification

(2p)

Given is the following figure.



C. Explain how plug-ins realize a variation mechanism in a product line (5p).

The plugin mechanism allows third components to be attached to the host application without having to know the exact internals. Hence, third parties can construct additional functionality for the host, and thus support variation (4P). As it is a software construct, it is a mechanism (1P).

Given is an excerpt of the table on variation mechanisms, with three variation mechanisms described.

Variation Mechanism		Properties building the Core Asset		Building the Product Variant		
Name	**Definition**	**Cost**	**Skills**	**Stakeholder**	**Tools**	**Cost**
Runtime conditionals	Selective execution of control flow	Medium	No special skills required	Product developer	None	None
Inheritance	Specializing or generalizing a particular class	Medium	Object Oriented Languages	Product developers	Compiler	Medium
Add-ons; Plug-ins	Software that enhances another software application and runs independently	High	Framework programming	End user	None	Low

D. Explain why Cost and Skills are important for "Properties building the Core Asset" (5p).

Cost: resources are needed to design, build and maintain the core assets. It should be worthwhile for core assets to add variation mechanisms. If the costs are too high one should not add the variation. (2P)

Skills: specific knowledge domains are needed to create the core asset, i.e., if the necessary skills for a certain mechanism are not present, one should not use that mechanism. (2P)

Both elements the architect needs to consider to make a proper assessment of which mechanism to use. (1P)

End of exam.