NTNU
Norwegian University of Science and Technology

Faculty of Informatics, Mathematics and Electronics

Department of Computer and Information Sciences

ENGLISH

Examination results will be announced: 16. June

Exam in the subject
TDT4240 Software Architecture

Monday 26. May 2008
9:00 am – 1:00 pm

Aids code C:

Simple calculator allowed.
These specified printed documents are allowed:

- IEEE (2000), "IEEE Recommended Practice for Architectural Description of Software-Intensive Systems", Software Engineering Standards Committee of the IEEE Computer Society.
- Kruchten, P. (1995), "The 4+1 View Model of Architecture", IEEE Software, 12(6).
- English-Norwegian dictionary (or to your native language if your not Norwegian) and/or a English thesaurus (English-English).

Contact person during the exam:

PhD-student Odd Petter N. Slyngstad, phone 7359 4483, mobile phone: 930 48 411

The points show how much each problem is worth in this exam. For each problem, each question has the same weight unless otherwise stated. The exam has 6 problems giving a total of 70 points. The remaining 30 points are credits awarded from the software architecture project.

Good Luck!

Salah Uddin Ahmed and Bian Wu

Controlled 8th of May 2008

Problem 1: Various questions (15 points)

Answer these questions in short:

1.1 What is Bass, Clements and Kazman's definition of Software Architecture (the definition in the textbook)?
1.2 What is the purpose of the CBAM architecture evaluation?
1.3 What is an idiom (related to design pattern)?
1.4 What is architectural drift?
1.5 Describe the Observer design pattern and advantages of using it.
1.6 Name five outputs from ATAM.
1.7 What is Attribute-Driven Design (ADD) according to the textbook (describe)?
1.8 Name the three main areas of performance tactics described in the textbook.
1.9 Explain how usability can be related to software architecture.
1.10 How can issues of the main stakeholders of a system be addressed in a software architecture documentation?
1.11 What is a reference architecture?
1.12 Explain how the architectural choices to obtain modifiability and performance in a system affect each other.
1.13 What is an architectural strategy according to the textbook?
1.14 What is the purpose of the IEEE1471?
1.15 Why can it be necessary to reconstruct a software architecture?

Problem 2: Service Oriented Architecture (SOA) (10 points)

2.1 Service Identification Framework (SIF) (5 points)

Illustrate, describe and explain the *Service Identification Framework* (SIF) and its main parts. Also describe the requirements that must be taken into account when the SIF is used.

2.2 Service components in SOA (5 points)

Illustrate and describe the *Service Component and its parts* as given in the Service Component Architecture.

Problem 3: Creating a game architecture (5 points)

Describe and explain the process of creating a game architecture according to the book "Game Architecture and Design" by Rollings & Morris.

Problem 4: ATAM (5 points)

Do the step 6 (Analyze the architectural approaches) in the ATAM process based on the following information about a system for selling tickets over the web:

Utility tree:

- *Availability:*
  - *Scenario A1. The system should have less than 1 minute of downtime in a week. (M,H).*
- *Security:*
  - *Scenario S1: 99.9% of all money transactions should be safe. (H,M).*

Identified architectural tactics:

- *Authenticate users* – use username and password to provide authentication.
- *Authorize users* – use access control patterns to give the approved rights to data or services to the authorized users.
- *Maintain data confidentiality* – apply encryption to data and communication links to protect data from unauthorized access.

Problem 5: Choose the correct architectural pattern (5 points)

Peter N. Erd is hired as a software architect in the Big Mess software project, where his job is to choose the correct architectural pattern for the system. The system should have the following qualities:

- It should be easy to replace components of the system also in run-time
- To ensure consistency of the system, all information should be maintained in one place
- If the data fails, the system should go back or stay in a consistent state

Help Peter to choose among the following architectural patterns and **motivate** your choice:

1. Pipe and filter
2. Layered
3. Blackboard
4. Task Control
5. NASREM

Problem 6 Create an architecture (30 points)

Read the description below and do the following:

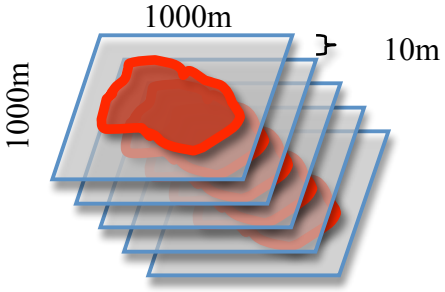
6.1 Identify the ***architectural drivers*** for the system described below (5 points)

6.2 Choose and describe suitable ***design and/or architectural patterns*** for the problem described below, and describe how the patterns affect the quality attributes (5 points)

6.3 Create ***architecture views*** of the system described below. The architecture must be described in two views according to the 4+1 view model: *Logical and Development view* (20 points)

Motivate for your choice of quality attributes, architectural drivers, design patterns and the architectural patterns used in your architecture. State your assumptions.

Software for an Oil Reservoir Computation System



The software described here is a system for performing computations related to oil reservoirs (areas in the ground where the oil is stored). The data input of the system is a large set of horizontal 2D-scans (2d-pictures) of the oil reservoir covering an area of 1km x 1km. A horizontal 2D-scan is produced for every 10m making it possible to make a 3D-representation of the reservoir.

Functional requirements:

- F1. Improve the horizontal 2D-scans by removing noise in the picture.
- F2. Improve the horizontal 2D-scans by sharpening the edges in the picture. This operation must be carried out after the noise has been removed.
- F3. Transform the horizontal 2D-scans to black & white picture, where the black area indicates oil and the white area indicates earth, rock or similar. The transformation is based on several parameters that maps colour values to what should be recognized as oil. This transformation must take place after noise removal.
- F4. Merge the pixels in the black and white horizontal 2D-scans that are close together to make solid graphical 2D-objects. This functionality is aimed at recognizing lungs of oil in the ground and removing additional noise in the picture. The result of the merge process is a picture that only consists of solid 2D-objects.
- F5. Transform a set of horizontal 2D-scans to a 3D-model representation. This operation will also take some parameters into account that will decide how the 3D-objects will be represented and how many 2D-scans should be input to the 3D-model.
- F6. Compute the total volume of oil from 3D-models.

User interface requirements:

- U1. The user interface must provide text boxes for the input parameters for the system that will change how the process is performed.
- U2. The user interface must show a 3D-visualisation of the oil reservoir.
- U3. The user interface must display the volume of oil found in the oil reservoir.

Quality requirements:

- M1. Modifiability: The various parts in the system must be possible to be replaced with parts that are improved, more efficient, or that uses other algorithms, etc.
- M2. Modifiability: The interfaces between the parts of the system should remain the same when the system is changed.