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: 8. June

Exam in the subject
TDT4240 Software Architecture

Friday 18. May 2007
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:

Associate professor Alf Inge Wang, phone 73594485, mobile phone: 92289577

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 5 problems giving a total of 70 points. The remaining 30 points are credits awarded from the robot project.

Good Luck!

Odd Petter Slyngstad

Controlled 8th of May 2007

Problem 1: Various questions (15 points)

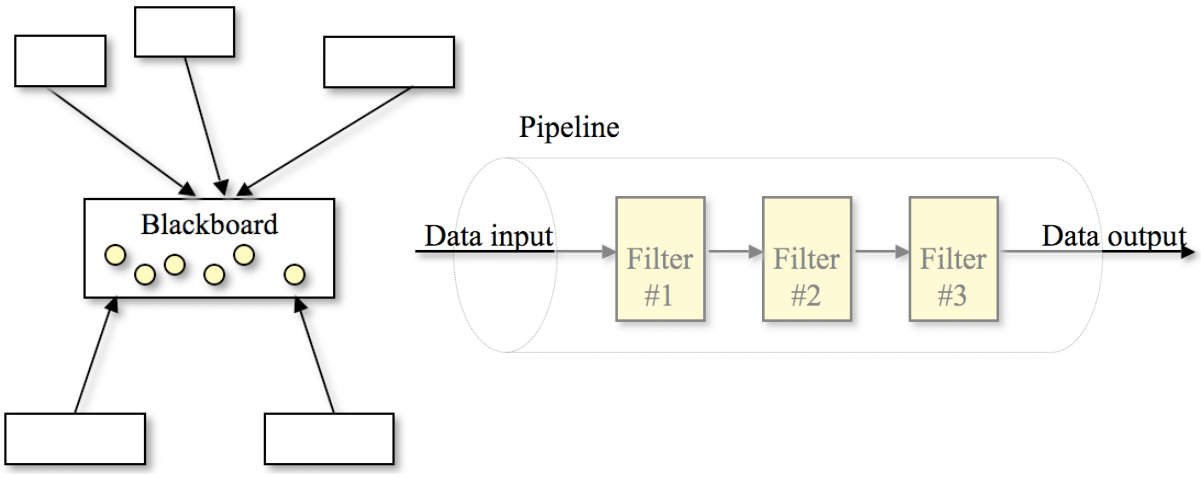
Answer these questions short:

1.1 What is Bass, Clements and Kazman's definition of Software Architecture (the definition in the textbook)?
1.2 How are quality attributes related to software architecture?
1.3 How can a software architecture benefit from using design patterns?
1.4 What is architectural erosion?
1.5 How can the architecture be influenced by how the company is organized (organization structures, departments, size, etc.)?
1.6 Describe the differences and the relationships between reference model, architectural pattern, reference architecture and software architecture.
1.7 Explain the difference between business quality, architecture quality and system quality according to the textbook (Bass, Clements and Kazman).
1.8 What is the relationship between tactics and architectural patterns?
1.9 What are the typical relations that you can extract when reconstructing software architecture (e.g. in the programming language C)?
1.10 What is the problem with too narrow or too broad scope for a software product line?
1.11 Name the main forces that affect a game architecture?
1.12 What is a service-oriented architecture?
1.13 What is a quality attribute scenario and what is the purpose of it?
1.14 What are the outputs of the ATAM?
1.15 Describe the Composite design pattern and its advantages?

Problem 2: Architectural patterns (10 points)

2.1 Compare two architectural patterns (5 points)

You are going to create the architecture for a robot controller where the main focus is on *modifiability*, so you easily can change the behaviour of the robot. ***Compare the two architectural patterns*** Blackboard and Pipe and filter (as shown below) and ***choose the pattern that best fits the quality requirement*** (motivate for your choice).



2.2 Describe the Subsumption Reference Architecture (5 points)

Describe the subsumption reference architecture using text and a figure, and describe the characteristics for this reference architecture.

Problem 3 (10 points): The CBAM

From the Table 1, 2 and 3 and the information below, find (use *straight lines between the data points in the graph*):

- Total benefit obtained from the 3 architectural strategies.
- Return-On-Investment for the 3 architectural strategies
- Rank the 3 architectural strategies according to best investment.

The data is results from applying the Cost Benefit Analysis Method (CBAM) on a web-system for selling tickets for cinemas, theatres, music concerts, and sports events.

Total Benefit Obtained can be computed using this formula: $B_i = \Sigma (b_{i,j} \times W_j)$ where:

- B is benefit for each architectural strategy *i*
- $b_{i,j}$ is the benefit by using strategy *i* to its effect on scenario *j*
- W_j is the weight of scenario *j*

Table 1: Results from prioritizing scenarios with worst, current, desired and best response levels.

Scenario	Vote	Worst	Current	Desired	Best
1. Performance (Time to pick up and deliver 4 balls to the light)	20	60min	20min	5min	3min
2. Availability (How many times the robot get stuck during the mission picking up balls)	20	50 % fail	20% fail	5% fail	0 % fail
3. Availability (How many times the robot controller software crashes)	35	30% crash	10% crash	0% lost	0% lost
4. Modifiability (Time to add readymade module)	15	8hours	60min	10min	5min
5. Testability (Time to test robot movements)	10	10hours	5hours	30min	10min

Table 2: Results from assigning utility to the various scenarios.

Scenario	Vote	Worst	Current	Desired	Best
1. Performance	20	10	50	90	100
2. Availability	20	10	55	85	100
3. Availability	35	15	75	100	100
4. Usability	15	0	50	70	100
5. Testability	10	35	60	80	100

Table 3: Effect and cost of using architectural strategies.

Strategy	Scenario	Cost	Current response	Expected response
1. Use route-planning	1	2000	20min	10min
2. Improved wall detector	2 3	800	20% fail 10% crash	7% fail 10% crash
3. Improved exception handling	2 3	1100	20% fail 10% crash	20% fail 2% crash

Problem 4: Reconstruction of a software architecture (5 points)

Read the description of the software system below and *suggest a practical approach for reconstructing the architecture of the system*. Describe the information you will use in the reconstruction and how you will use it.

The company HugeSoft™ implemented the banking system BankBankWhoIsThere (BBWIT) in 1986 and several banks have used this system since then. HugeSoft wants integrate this system with an insurance system and need to establish a software architecture to see if it is possible and how. No software architecture or design documentation exists for the system, and none of the architects or designers of the system are accessible. The only artefacts you have available from the system are the 8,000 files of source code written in the programming language C distributed in more than 200 directories. The system consists of more than 2,000,000 lines of source code, and most parts of the system are divided into modules that usually include hundreds of files.

Problem 5 Create an architecture (30 points)

Read the description below and do the following:

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

5.2 Choose and describe suitable ***architectural tactics*** for the problem described below, and describe how the tactics affect the quality attributes (5 points)

5.3 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 Process view* (20 points)

Motivate for your choice of quality attributes, architectural drivers and the architectural tactics used in your architecture. State your assumtions.

Software for SoNewHenriksen$_{TM}$ Mobile Phones

The software described here is software for operating various models of SoNewHenriksen$_{TM}$ (SNH) mobile phones. The different models vary in the type of data communication supported, display, keypad and buttons, processor and the type of functionality offered. A SNH mobile phone consists of the following physical parts:

- A mobile processor with memory, Flash ROM and storage (varies from model to model).
- A display (varies from model to model)
- Buttons and keypads (varies from model to model)
- A communication unit responsible for all communication (varies from model to model)
- A loudspeaker (same for all models)
- A microphone (same for all models)
- A battery (same for all models)
- A connection interface for charging and connection to other units like PCs (same for all models)

The software for operating the SNH mobile phones provides the following basic services that are the same for all phone models:

- Call-functionality (e.g., make phone calls, list of callers etc.).
- Address book.
- Messaging functionality (e.g., SMS, MMS, Email, etc.).
- Time, Calendar and Alarm.
- Setting/Preferences.
- File management.

In addition SNH mobile phones can offer additional functionality (varies from model to model) through addition applications that can be pre-installed on the phone:

- Internet functionality (e.g., web-browser).
- Multimedia functionality (e.g. audio player, video player, radio, etc.)
- Camera functionality.
- Games.
- Etc.