SEN 3244

SOFTWARE ARCHITECTURE

M. Mangong Clement

❑ Chap1: Introduction to Software Architecture

Lesson Objectives

At the end of this lesson, we will answer the questions:-

- What is architectural?
- Why evaluate an architecture?
- When can an architecture be evaluated?
- Who's involved?
- What results does an architectural evaluation product?
- What are the output of an architecture evaluation?

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

Answering question(1) as…

- **How can you be sure whether the architecture chosen for a software is the right one?**

Its not an easy question !!!

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

Answering question(2) as…

- **How can you be sure that it won't lead to calamity but instead will pave the way through a smooth development and successful product?**
- ***Are not easy questions !!!***

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

- ➢ **The foundation for any software system is its architecture.**
- ✓ **The architecture should allow all of a system's quality attributes:- such as modifiability, performance, security, availability, reliability- all of these are precast once the architecture is laid down.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

- **Waiting until the system is completed before knowing whether it will meet its requirements (or not) is not a good option.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

- If you're buying a software or paying for a software development, wouldn't you like to have some assurance that its started off down the right path?
- If you're the architect yourself, wouldn't you like to have a good way to validate your initiative and experience?

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

Architecture evaluation method include:-

- ➢ **ATAM: Architecture Tradeoff Analysis Method**
- ➢ **SAAM: Software Architecture Analysis Method**
- ➢ **ARID: Active Reviews for Intermediate Designs**
- Evaluations represent a wise risk-mitigation effort and are relatively inexpensive.

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

In the following slides,

- ***We shall define what we mean by software architecture***
- ***And explain the kinds of properties for which an architecture can (and cannot) be evaluated.***

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

What is software architecture of a program?

- ***The structure or structures of the system, which comprise software components, the externally visible properties of those components, and the relationships among them.***

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

By externally visible properties …

- ***The assumptions other components can make of a component, such as its provided services, performance characteristics, fault handling, shared resources usage, and so on.***

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

- **An architecture defines the components such as modules, objects, processes , subsystems, compilation units, and so forth.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

- **An architecture defines the relevant relations such as calls, sends-data-to, synchronizes-with, uses, depends-on, instantiates, and many more.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

- ➢ **The architecture is the results of early decisions that are necessary before a group of people can collaboratively build a software system.**
- ✓ **The larger or more distributed the group, the more vital the architecture is**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

- ➢ **Architecture allows nearly all of the system's quality attributes.**
- ✓ **Does this leads to the fundamental truth about architecture evaluation?**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

✓ **This leads to the question, if architectural decisions determined a system's quality attributes, then it is possible to evaluate architectural decisions with respect to their impact on these attributes?**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

✓ **What is architectural?(1)**

➢ **Those evaluating architecture have a pressing need to understand what information is in and out of their realm of concern.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

- ✓ **What is architectural?(1)**
- ➢ **Some may ask the question in the following way:-**
- ✓ **What is the difference between an architecture and a high-level design?**
- ✓ **Are details such as priorities and process architectural?**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

- ✓ **What is architectural?(2)**
- ✓ **Are interfaces to components part of the architecture?**
- ✓ **If I have class diagrams, do I need anything else?**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

- ✓ **What is architectural?(3)**
- ✓ **Is architecture concerned with run-time behaviour or static structure?**
- ✓ **Is the operating system part of the architecture? Is the programming language?**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

- Firstly, our software architecture definition tells about components, their externally visible properties, and the relationship among them, but it fails to address the notion of context.
- Therefore context influence architecture

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ **Secondly , what is not architectural?**

The criterion for something to be architectural is:-

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

- ✓ It must be a component, or a relationship between components, or a property that needs to be externally visible in order to reason about the ability of the system to meet its quality requirements or to support decomposition of the system into independently implementable pieces.

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ Why evaluate an architecture?

- ✓ The earlier you find a problem in a software project, the better off you are.
- ✓ The cost to fix an error found during requirements or early design phases takes less time to correct than the same error found during testing.

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ Why evaluate an architecture?(1)

An unsuitable architecture will precipitate disaster on a project.

- ✓ **Performance goals will not be met**
- ✓ **Security goals will fall by the wayside**
- ✓ **The customer will grow impatient because the right functionality is not available,…**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ Why evaluate an architecture?(2)

- Architecture evaluation is a cheap way to avoid disaster.

✓ When building a house, you wouldn't think of proceeding without carefully looking at the blueprints before construction began.

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ When can an architecture be evaluated?

- **The classical application of architecture evaluation occurs when the architecture has been specified but before implementation has begun.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ When can an architecture be evaluated?

- **Users of iterative or incremental life-cycle models can evaluate the architecture decisions made during the most recent cycle.**
- **However, it can be applied at any stage of a architecture's lifetime**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ Who's involved?

- The are two groups of people involved

1. **Evaluation team :** People who will conduct the evaluation and perform the analysis.
2. **Stakeholders :** People who have vested interest in the architecture and the system that will be built from it. E.g project decision maker

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ What result does an architecture evaluation produce?

- **It produces a report; the report form and content vary according to the method used.**
- **The information produce give answers to two kinds of questions.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ What result does an architecture evaluation produce?(1)

- **The information produce give answers to two kinds of questions.**

1. ***Is this architecture suitable for the system for which it was designed?***
2. ***Which of two or more competing architectures is the most suitable one for the system at hand?***

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ What result does an architecture evaluation produce?(2)

- **Suitability of the designed architecture is what we seek to investigate.**
- **We say that an architecture is suitable if it meets two criteria.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ What result does an architecture evaluation produce?(3)

- **Two criteria of an architecture suitability**

1. **The system that results from it will meet its quality goals. (performance requirements, …)**
2. **The system can be built using the resources at hand: the staff, the budget, legacy software,…**
   Is the architecture is buildable.

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ What result does an architecture evaluation produce?(4)

- **Suitability is only relevant in the context of specific goals for the architecture and the system.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ What result does an architecture evaluation produce?(5)

- **An architecture designed with high-speed performance as the primary design goals might lead to a fast system but may require programmers to work for a month to make any kind of modification.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ What result does an architecture evaluation produce?(6)

- **If modifiability were more important than performance for that system, then that architecture would be unsuitable for that system.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ What result does an architecture evaluation produce?(7)

- **If the sponsor of a system cannot tell you what any of the quality goals are for the system, then any architecture will do.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ What result does an architecture evaluation produce?(8)

- **An architecture evaluation doesn't tell you "yes" or "no", "good", or "bad" or "6.75 out of 10". It tells you where you are at a risk.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ What result does an architecture evaluation produce?(9)

- **Architecture evaluation can be applied to a single architecture or to a group of competing architecture.**
- **An architecture evaluation reveals the strengths and weaknesses of each one.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ What result does an architecture evaluation produce?(9)

- **The evaluation will first identify what the areas of interest are and then highlight the strengths and weaknesses of each architecture in those areas.**
- **Management must decide on the competing architecture should be selected or improved.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

- ➢ For what qualities can we evaluate an architecture?

Can you look at an architecture and determine if the architecture will meet all the quality goals?

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ For what qualities can we evaluate an architecture?(1)

Case 1

- **An implementation might diverge from the architectural plan that compromise the quality plan.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ For what qualities can we evaluate an architecture?(2) Case 2

- **An architecture does not strictly determine all of a system qualities.**

✓ **Example: Usability is the measure of a user's ability to utilise a system effectively and important quality goals for many systems but is largely a function of the user interface.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ For what qualities can we evaluate an architecture?(3)

- **Most user interface aspects like whether the user sees red or blue backgrounds, a radio button are by a large not architectural since those decisions are generally defined to the lower level of design.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ For what qualities can we evaluate an architecture?(4)

- **Other quality attributes lies squarely in the realm of architecture.**
- **Example : ATAM evaluation method concentrate on evaluating an architecture for suitability in terms certain quality attributes.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ For what qualities can we evaluate an architecture?(5)

- **Example : ATAM evaluation method**

✓ **Performance – ?**

✓ **Reliability – ?**

✓ **Availability -?**

✓ **Functionality -?**

✓ **Conceptual integrity-?**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ For what qualities can we evaluate an architecture?(6)

- **Example : ATAM evaluation method**

✓ **Security – ?**

✓ **Modifiability -?**

✓ **Portability – ?**

✓ **Variability - ?**

✓ **Subsetability-?**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ For what qualities can we evaluate an architecture?(7)

- **Example : For SAAM evaluation method**

✓ **Modifiability in its various forms (such as portability, subsetability and variability) and functionality.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ For what qualities can we evaluate an architecture?(8)

- **Example : For ARID evaluation method**

✓ **Provides insights about the suitability of a portion of the architecture to be used by developers to complete their tasks.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ Why are quality attributes too vague for analysis?

- Quality attributes form the basis for architectural evaluation.
- But simply naming the attributes by themselves is not a sufficient basis on which to judge an architecture for suitability.

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ Why are quality attributes too vague for analysis?(1)

- **Requirement statements are often like the following**

✓ **"The system shall be robust"**

✓ **"The system shall be highly modifiable"**

✓ **"The system shall be secure from unauthorised beak-in"; *Give other examples…***

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ Why are quality attributes too vague for analysis?(2)

- Without elaboration, each of these statements is subject to interpretation and misunderstanding.
- Qualities attributes exist in the context of specific goals.

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ Why are quality attributes too vague for analysis?(3)

- A system is modifiable (or not) with respect to a specific kind of change
- A system is secure (or not) with respect to a specific kind of threat
- A system is reliable(or not) with respect to a specific kind of fault occurrence.

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ Why are quality attributes too vague for analysis?(4)

- A system is suitable well (or not) with respect to specific performance criteria
- A system is suitable(or not) for a product line with respect to a specific set or range of envisioned product in the product line

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ Why are quality attributes too vague for analysis?(5)

- **Using scenario is one of the best ways get the quality for evaluation.**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ What are the outputs of an Architecture Evaluation?(1)

- **Output from the ATAM, the SAAM and ARID produces the output**

a) Prioritized statement of quality attribute requirements

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ What are the outputs of an Architecture Evaluation?(2)

- **Output from the ATAM, the SAAM and ARID produces the output**

b) Mapping of approaches to quality attributes

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ What are the outputs of an Architecture Evaluation?(3)

- **Output from the ATAM, the SAAM and ARID produces the output**

c) Risk and nonrisks

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ What are the outputs of an Architecture Evaluation?(4)

- **The output of each of them shall be treated in the next lesson**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ What are the benefits and costs of performing an architecture evaluation?

- **Main benefits is to uncover problems that if left undiscovered would be orders of magnitude more expensive to collect later.**
- **It produces better architecture**

❑ Chap1: Introduction to Software Architecture

❖ Evaluating a software Architecture

➢ What are the benefits and costs of performing an architecture evaluation?

- **We shall elaborate on the various benefits during out next class.**

Welcome!

This course is design for you to understand the ways software architectures are represented, both in UML and other visual tools.

QUESTIONS