Politecnico
di Milano

The design process

Slides used in the video available here
https://polimi365-my.sharepoint.com/:v:/g/personal/10143828_polimi_it/EcbU-pgqCblMiw2hIZIUFSABffi1zSZCjXuYrgt1cQfntw?e=ebYZC9

The Process of Architectural Design

- Definition:
  - *Defining the software architecture or design* is a problem-solving process whose objective is to find and describe a way:
    - To implement the system *functional requirements*...
    - While respecting the constraints imposed by the *quality, platform and process requirements...*
      - including the budget
    - And while adhering to general principles of *good quality*

Top-down and bottom-up design

- Top-down design
  - First design the very high level structure of the system.
  - Then gradually work down to detailed decisions about low-level constructs.
  - Finally arrive at detailed decisions such as:
    - the format of particular data items;
    - the individual algorithms that will be used.

Top-down and bottom-up design

- Bottom-up design
  - Make decisions about reusable low-level utilities.
  - Then decide how these will be put together to create high-level constructs.

- A mix of top-down and bottom-up approaches are normally used:
  - Top-down design is almost always needed to give the system a good structure.
  - Bottom-up design is normally useful so that reusable components can be created.

Top-down vs. bottom-up

A top-down design

- A very nice end result!
- It will take time to get it right…
- Individual components harder to re-use

A bottom-up design

- A bit rough end result
- It will be quicker to get to…
- Individual components easier to re-use

You are to find the right trade-off!

Design as a series of decisions

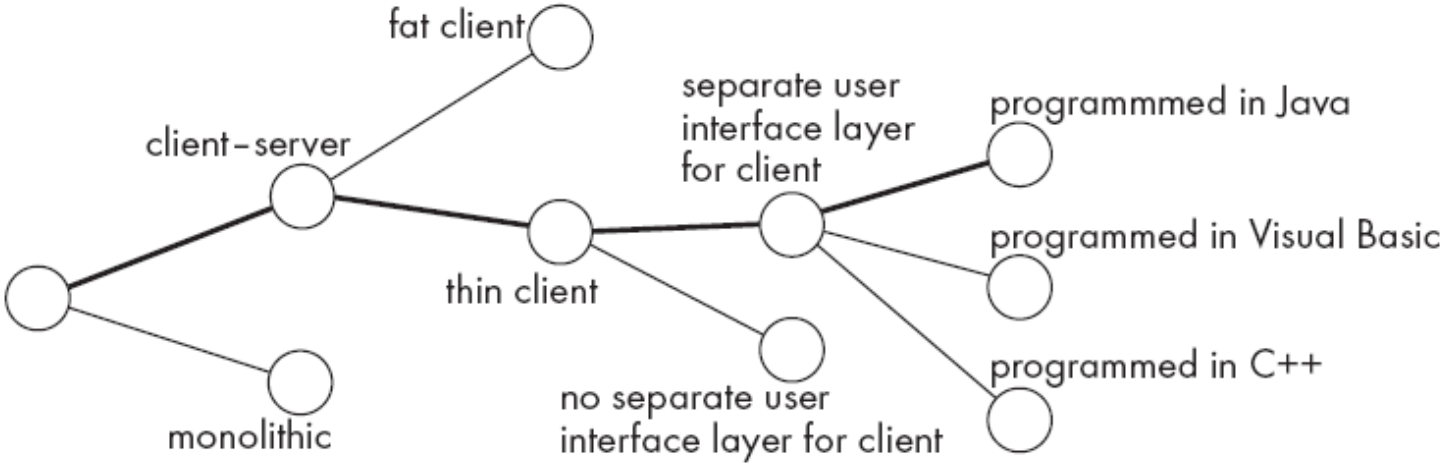
- A designer is faced with a series of *design issues*
  - These are sub-problems of the overall design problem.
  - Each issue normally has several alternative solutions:
    - Design *options*.
  - The designer makes a *design decision* to resolve each issue.
    - This process involves choosing the best option from among the alternatives.

Making decisions

- To make each design decision, the software engineer uses:
  - Knowledge of
    - the requirements
    - the design as created so far
    - the technology available
    - software design principles and ‘best practices’
    - what has worked well in the past

Design space

- The space of possible designs that could be achieved by choosing different sets of alternatives is often called the design space
  - For example:

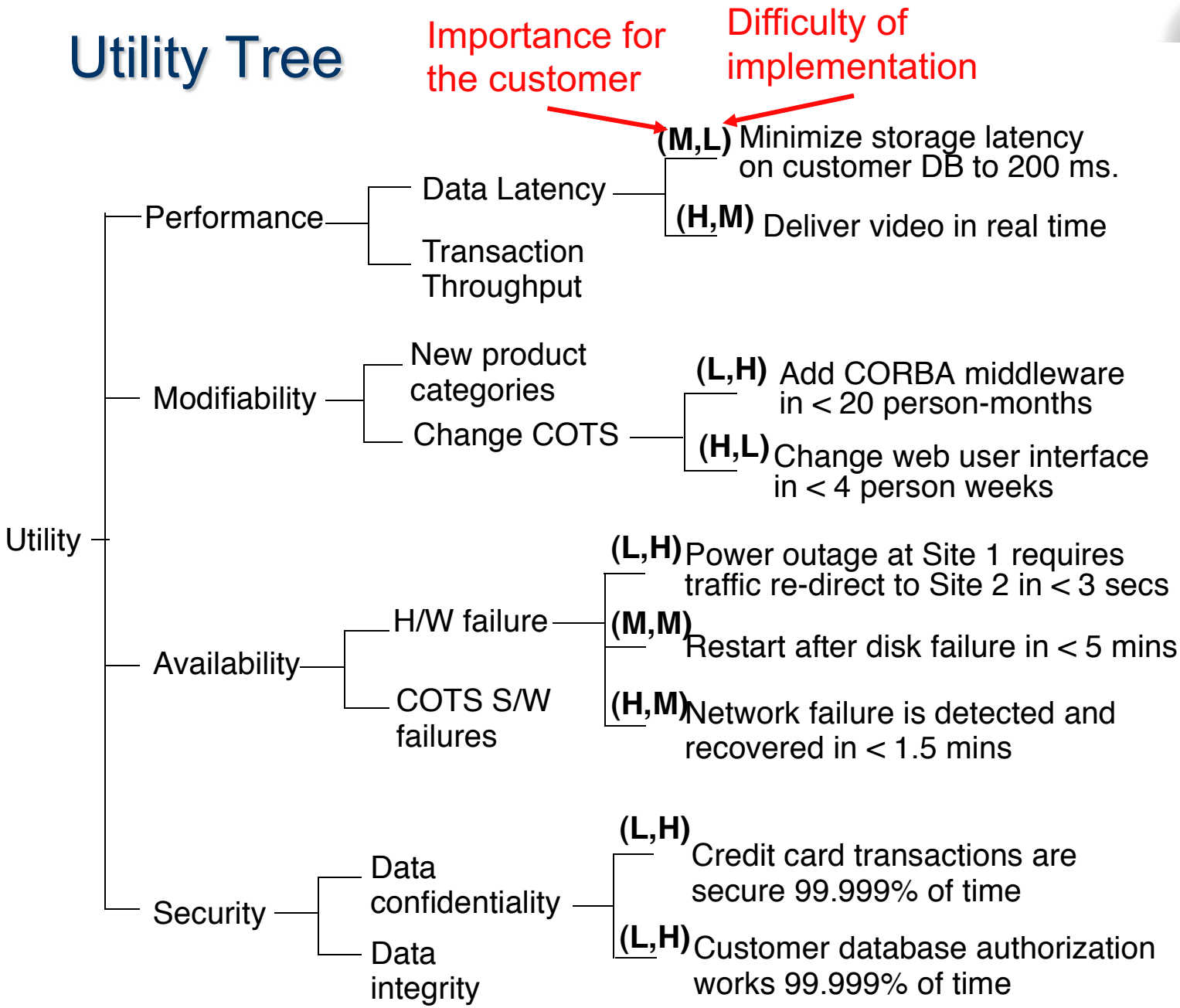


ATAM: the Architecture Tradeoff Analysis Method

https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5177

- It supports making the consequences of design decisions explicit
- It makes possible for stakeholders to trade off the different possibilities, and make informed decisions, with clear insight into the consequences thereof
- It supports users to determine how quality attributes interact.
  - Example1: If we decide to include an authorization component to increase security, this is likely degrading performance.
  - Example2: If we add redundancy to increase availability we increase cost and (possibly) degrade performance.

Utility Tree



From identification to quality assessment

- **ATAM Investigation and Analysis phase**
  - 1. Identify architectural approaches (styles): Architectural approaches supporting the system and business goals are identified by the architect, but not analyzed
  - 2. Generate quality attribute utility tree: Quality factors that comprise system "utility" (performance, availability, security, etc.) are elicited, specified down to scenarios level, annotated with stimuli and responses, and prioritized.
  - 3. Analyze architectural approaches: The stakeholders and the architect analyze how the architectural approaches affect to the quality factors identified in Step 2.

Questionnaire

- To check whether you have got the main points in this video, please fill in the questionnaire you find here
  - https://forms.gle/d2eHhiPT2vySFyQu5
- Your answer will not be used for assessment of your performance