



Documenting Architectures

Mikael Svahnberg¹

2018-02-01

¹Mikael.Svahnberg@bth.se



Outline



Purpose

- Education
- Communication
- Initiater Discussions
- Support Analysis and Evaluations
- Early Design Decisions
- Resource Allocations
- Support Maintenance

Documentation

- Abstract enough to understand
- Detailed enough to analyse
- Describes constraints
- Recounts Decisions



Views

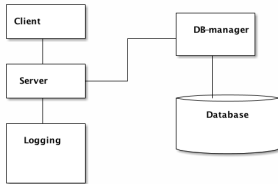
- Fixed Set of Views (Kruchten 1995, UML, Hofmeister et al.)
- Flexible set of views (Bass et al.)
 - Choose relevant views *for the current system*
 - Document the view
 - Behaviour
 - Interfaces
 - Document information that applies to more than one view.



Different Documentation Techniques

- Informal (boxes and lines)
- Structured Diagrams
- Formal Specifications
- Architecture Description Language (ADL)

Boxes and Lines



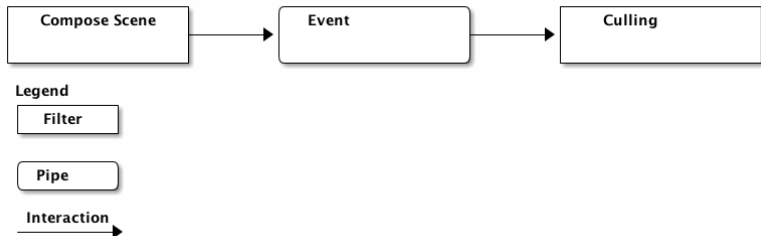
Questions that need answers:

- What is the nature of the boxes?
- Do the boxes have similar behaviour?
- What is the significance of the lines?
- How does the behaviour of the parts contribute to the behaviour of the system?
- Does the layout have any meaning?
- Is the sketch realisable?



Structured Diagrams

- Boxes and Lines, but with *semantics* in the form of a legend.



- Some methods may come with a notation.
- For example, Hofmeister et al. uses extended UML.
- Consistent Use is more important than choice of Notation

UML Package Diagrams

- Package Diagrams can be used to represent different Views, typically *Conceptual*, *Module*, and *Execution* views.
- No obvious guidelines for *how*, but
 - Pick one style per view, and stick to it.
 - Reserve the “default” view for a *Module view*.
- The nature of Interactions still need to be defined

Examples:

