

Architectural Pattern

Thomas Traxler

May 23, 2014

Contents

1	Architecture	3
---	--------------	---

1 Architecture

"The word conjures up images of flying buttressed cathedrals and Doric columns, even among computer cognoscenti. Yet more than years have passed since computer science appropriated the term for a new meaning. Back then most people thought 'software development' meant 'writing code', but some in the software development community knew better. They observed people attempting ever-larger software systems, and they saw how traditional ad hoc development approaches were failing them with alarming regularity.

'Large' systems are inherently larger than one person can build. So to build them, developers had to foster discipline and cooperation. They had to scope out their system ahead of time and break it up into manageable pieces. They needed ways of specifying what the pieces did and how they communicated with other pieces. They had to define abstractions that gave programmer and maintainers insight into the system's workings. They had to be creative in the large, not just clever in the small. In short, they needed a notion of overall design apart from implementation - *software architecture* "[LS96]

I wanted to start with this cite of the beginning of the Chapter 'Architectural patterns' from the book pattern languages of program design 2 because it describes the problematic in nowadays software development in combination with the term *Architecture* in it's classic meaning as good as nobody else.

Bibliography

- [LS96] LAVENDER, R G. ; SCHMIDT, Douglas C.: Pattern Languages of Program Design 2. In: *Pattern Languages of Program Design 2*. 1996. – ISBN 0-201-895277, S. 483–499