Brady Field

Architectural Styles, Design Patterns, and Objects

**Important points**

“Architectural designs illuminate such key issues as scaling and portability, the assignment of functionality to design elements, interaction protocols between elements, and global system properties such as processing rates, end-to-end capacities, and overall performance.” [249]

“These convey widespread if informal understanding of the descriptions and let engineers quickly communicate their designs to others.” [250]

“An important class of architectural idioms constitutes what some researchers have termed architectural styles. An architectural style characterizes a family of systems that are related by shared structural and semantic properties.” [251]

“The object-oriented design paradigm provides another abstraction for software design. In its simplest form, an OOD lets system designers encapsulate data and behavior in discrete objects that provide explicit interfaces to other objects.” [252]

“Any object that can send a message to another object can request that the target object invoke any of its public methods. There is effectively a single, flat interface provided by all objects to all objects.” [253]

“An object-oriented approach to specifying an architectural pipe connector for use in pipe-and-filter style systems, along with rules for how a pipe can be properly instantiated in a design, apparently will require the cooperation of multiple objects.” [254]

“Architectural design is concerned with composing systems from components, and the interactions between these components. Such compositions provide an abstract view of a system, so that the designer can do system-level analyses and reason about system integrity constraints.” [255]

“We see patterns and architectural styles as complementary mechanisms for encapsulating design expertise. An architectural style provides a collection of building-block design elements, rules and constraints for composing the building blocks, and tools for analyzing and manipulating designs created in the style.” [256]

“Thus, architectural design patterns and object-oriented design patterns are simply instances of the more general class of all design patterns.” [257]

“Architectures, architectural styles, objects, and design patterns capture complementary aspects of software design. Although the issues and aspects of software design addressed by these four approaches overlap somewhat, none completely subsumes the other. Each has something to offer in the way of a collection of representational models and mechanisms.” [258]

**Disagreements**

“First, architectural styles can be viewed as kinds of patterns – or perhaps more accurately as pattern languages. Describing an architectural style as a design pattern requires, however, a rather broad definition of the scope of design patterns.” [256]

The authors first assert that architectural styles are a subset of patterns. Immediately following, the authors then backtrack and point out that they are not actually the same thing. We can see this in the next line.

“An architectural style is probably better thought of as a design language that provides architects with a vocabulary and framework with which they can build useful design patterns to solve specific problems – much as OMT provides a framework and notation for working with objects.” [256]

The authors further contradict their initial statement later on by describing these two terms as complimentary.

“We see patterns and architectural styles as complementary mechanisms for encapsulating design expertise. An architectural style provides a collection of building-block design elements, rules and constraints for composing the building blocks, and tools for analyzing and manipulating designs created in the style.” [256]

It is clear that the authors do not view these two things as the same at all. The initial sentence should be revised to agree with the points that follow.

**Questions**

I have no questions. I understood everything in the article.