

GOF Patterns Applied

Summary

Design Patterns are proven and powerful techniques that can help improve the resiliency, maintainability, and extensibility of your applications. However, overusing or misapplying patterns is a common mistake often times resulting in applications that are over-architected, and resemble a tangled web of classes. How can patterns be applied to achieve the goal of better software?

This session offers a gentle introduction to design patterns by examining a small, yet very usable framework designed using 7 common GOF patterns. We'll examine the framework, and explore how each pattern is used as well as how each pattern emerged based on real, instead of perceived need. Less flexible alternatives to the chosen patterns will also be discussed. To conclude, we'll examine how the patterns are applied in conjunction with each other, forming more complex compound patterns. Each pattern will be presented with accompanying source code, and all examples are gleaned from real world scenarios.

Audience

GOF Patterns Applied will benefit **software architects** and **software developers**. Through numerous examples using a small framework, we will examine how design patterns can be used during initial design and through refactoring to produce more extensible and resilient code.

Content Outline (90 minutes)

1. Introduction (15 minutes)

Goal: Review design pattern terminology and introduce GOF patterns.

Topics Include:

- Pattern Definition
- Introduction to GOF Patterns

2. Framework Case Study (60 minutes)

Goal: Present a small framework developed using patterns. Explore how and why various patterns were chosen, as well as possible alternatives to the approach selected. Upon completion, a usable framework will have been developed with multiple interwoven patterns offering greater framework flexibility.

Topics Include:

- Command Pattern
- Singleton Pattern
- Decorator Pattern
- Strategy Pattern
- Mediator Pattern
- Observer Pattern
- Builder Pattern

3. Pattern Retrospective (15 minutes)

Goal: Examine the future of patterns and explore the ideas behind compound patterns, heuristics, and the fundamental characteristics of object orientation common to most patterns.

Topics Include:

- Gleaning Heuristics from Patterns

- Compound Patterns

Presenter Background

Kirk is the Senior Technology Strategist at TeamSoft, Inc. (<http://www.teamsoftinc.com>), where he leads based on his firm belief in the pragmatic use of technology. In addition to his work on enterprise development projects, Kirk shares his experiences through courseware development and teaching, writing, and speaking at seminars and conferences. Kirk has provided training and mentoring to thousands of software professionals, teaching courses on object-oriented development, Java, software architecture, software process, and UML.

Kirk is the author of Java Design: Objects, UML, and Process, and the founder of www.extensiblejava.com, a growing resource of design pattern heuristics that emphasize greater component modularity in large scale enterprise software projects. He is a frequent contributor to The Agile Journal, where he writes The Agile Developer column. He is also the creator of JarAnalyzer and AssAnalyzer, utilities for identifying and managing the physical dependencies between Java .jar files and .Net dll assemblies, respectively. files. His personal website is www.kirkk.com, his blog is <http://techdistrict.kirkk.com>, and his planet is <http://planet.kirkk.com>.

Tutorial History

The material for this session was gleaned from a production framework still in use today. This session was given at Software Development Best Practices, and received greater than 90% approval rating. After the session, one attendee comment that this was the best presentation on patterns he'd ever attended. Since then, it has been presented at numerous other conferences and user groups, and is also presented on the NoFluffJustStuff Software Symposium tour.

Presenter Contact Information

Kirk Knoernschild
email: pragkirk@kirkk.com