Title

"Practical Polymorphism: Suggestions for Authoring High-Quality and Extensible Object-Oriented Software"

Abstract

Most modern software applications are developed using object-oriented programming (OOP) techniques. Great software developers understand that there is more to OOP than simply grouping data and methods into classes and objects. When should you encapsulate? When should you derive? Should you use public inheritance or private inheritance? When should you override a function from a base class? What are the pitfalls of doing so? How do you best use OOP to make your software more consistent, more readable, more reusable, and more reliable? This talk discusses the answers to these questions and more, and draws from real-world experience in a large codebase that has been in existence for more than twenty years.

Biography

Chris Tenbrink is a Senior Software Engineer for Chief Architect Software in Coeur d'Alene, Idaho. He has had a love for geometric modeling and computer-aided design software since he first used a CAD system as a young engineering student. Over the past decade, he has contributed to multiple releases of Chief Architect's industry-leading home design software. His experience at Chief Architect has helped him develop a passion for well-designed software that is extensible, maintainable, and robust. Currently, he is applying that passion as team lead for Room Planner, Chief Architect's home design app for mobile devices. He holds a B.S. in Mechanical Engineering from Brigham Young University and an M.S. in Mechanical Engineering from Purdue University.