

Writing High Quality Code

This 90-minute free webinar (with an additional 30 minutes at the end for Q&A) will help you quantify software qualities. Quality in delivered software is intangible and very different from quality in physical goods. Some external attributes of quality software—free from defects and easy to maintain—are reflections of the code’s internal qualities. When classes and methods are cohesive, non-redundant, well-encapsulated, assertive, and explicitly coupled, they are less prone to mistakes and far easier to debug, test, and maintain. David Bernstein asserts that paying attention to code quality helps us focus on the key principles, patterns, and practices used by expert developers. If you don’t pay attention to critical code quality attributes, iterative development practices can quickly degrade code into a maintenance nightmare. Join David and take a deep dive into the code qualities that make software more maintainable and less bug-friendly.

Agenda

- **What is Quality?** - Software Quality is Different; Correctness Isn’t Enough; Maintainability; Dealing with Change; Code Pathologies; Larger Projects are Harder; Quality in Software; Code Qualities
- **Cohesive** - What is Cohesion? Language and Software; Class Cohesion; Method Cohesion; Pathologies of Poor Cohesion
- **Non-Redundant** - Finding Redundancy; Type is Redundancies; Code Investment or Code Debt; Pathologies of Redundancy
- **Encapsulated** - What Can Be Encapsulated? What You don’t Know Can’t Hurt You; Encapsulate by Policy; Optimizing for Performance; Can You Encapsulate it? Many Ways to Encapsulate; Pathologies of Poor Encapsulation
- **Assertive** - Being Assertive; Assertive or Inquisitive; Prefer Assertiveness; Pathologies of Inquisitiveness
- **Testable** - Code Must Be Testable; Testability Guiding Design; Pathologies of Poor Testability
- **Explicit** - What are Explicit Relationships? Good Coupling and Bad Coupling; Kinds of Coupling; Coupling Example; Pathologies of Bad Coupling
- **Conclusion** - The Six Code Qualities; CREATE Quality Software; Code Qualities Guide Us



Your Instructor



My continuing passion for software design and construction has led me to train more than 10,000 professional software developers for clients that have included Fortune 500 firms such as Microsoft, IBM, Yahoo, Boeing, AT&T, Sprint, Medtronic, SunGard, State Farm, Vanguard, and Weyerhaeuser. As a longtime IBM consultant, I trained software engineers around the globe, giving them the skills to write the next generation of applications and operating system software while earning one of the highest satisfaction ratings in the history of IBM education. Since 2006, I’ve devoted my consulting practice to providing organizations with training and coaching for software developers and teams transitioning to Agile, Scrum, and Extreme Programming practices.