

The Refactory, Inc.

Overview

Contact Info

Joseph W. Yoder
The Refactory, Inc
7 Florida Drive
Urbana IL, 61801
(217) 344-4847
joe@refactory.com

Overview

The Refactory, Inc. aims to improve the quality of life for those involved with all aspects of software development. To realize this, we engage our clients through dialogue and education to help illuminate new paths to better solutions. The Refactory does this by providing top-notch consulting and training services to our clients and their software development teams.

The Refactory bases our mentoring and consultation on proven practices backed by industry leading research. Through our combined years of industry and academic experience, our consulting services aim to put ourselves out of business by uplifting our clients to a point where they no longer need us.

The Refactory, Inc. provides many different services that cover all different aspects of the software development process. We specialize in everything that deals with software development, including analysis and design, architecture, agile methods, building frameworks, refactoring, testing, and eXtreme programming. We offer on-site and public consulting, mentoring, training, and programming.

Analysis and Design

Analysis and Design are important aspects in understanding the evolution of existing systems and the development of new ones. The Refactory, Inc. principals are some of the most well respected framework developers, object-oriented designers, and pattern writers in the field and each have many years of experience with analysis and design, specifically object-oriented systems. We specialize in light weight methodologies, specifically those now being recognized as Agile Software Development.

The Refactory Inc. advocates a practical balance of analysis and design which integrates an understanding of system requirements as an organic part of the software development process. Maintaining this balance can prevent a tendency toward over-analysis, which, when take to extremes can lead to "analysis paralysis." The Refactory specializes in the development of quality systems through the use of frameworks and refactoring, using light-weight programming methodologies (agile practices), testing (pragmatic TDD), refactoring, and adaptive object models. Our agile methods increase production speed, while maintaining adaptability and ensuring quality systems.

Architecture

As computer and software systems evolve more complexity due to changing business and home user requirements, effective software design becomes more important than ever. As software systems evolve, they often become less manageable, as simple expedient solutions are implemented with no concern for the overall function of the application. Over time this eventually leads to a system which needs to be completely refactored or redesigned. While this cycle is not always inevitable, it's process can be substantially slowed, if not prevented, with the use of proper software architecture.

Often, proper architecture is deemed an unnecessary expenditure in today's rushed business world. Deadlines and bottom lines compete with quality for importance. Quality, efficient software architecture can be expensive, but "If you think good architecture is expensive, try bad architecture." (Big Ball of Mud, Foote, Yoder)

The Refactory specializes in developing and adapting frameworks and implementing analysis-and-design best practices for dynamic architecture. We can perform an assessment of your current system, offering suggestions, consultation and training to your software development team for evolving your current architecture, cleaning up "muddy" systems or designing new systems.

Consulting and Mentoring

We find consulting and mentoring to be an incredible opportunity for growth and understanding within the field. Our principal members have both consulted and mentored extensively within the commercial world to bring a consciousness about frameworks, objects, patterns, adaptive systems, and other groundbreaking research to life. We believe that the sharing of information is an essential part of the software development process. Our principal members share this information with one another, building what is essentially on the other's ideas and vice versa. This enables us to competitively keep our methods ahead of the field standard.

Ralph Johnson is one of four authors of the best selling book "Design Patterns" and has been dealing with objects since their inception; Don Roberts is an expert with analysis, design and software development - specifically OO systems; Don Roberts and John Brant are co-authors of the first commercial refactoring tools; Joseph Yoder has taught, mentored, consulted, and managed various software projects for many years, including applying patterns, designing architecture, creating adaptive object models, utilizing agile methods, developing frameworks, refactoring existing systems, object oriented software development in Java, C#/.NET, Smalltalk and C++, and web design and cloud services.; Brian Foote is the author of over two-dozen published patterns and has been working with frameworks, patterns, and objects for years.

The principals have experience with many companies, including CAT, Iron Mountain Fulfillment Services, Ernst & Young, Hewitt Associates, Illinois Department of Public Health, Illinois Department of Insurance, Lucent Technologies, North American Lighting, American Express Financial Advisors, Lifespan Research Institute, Industrial Logic, Communication Security Establishment of Canada, IBM, NCSA, Sprint, Ralston-Purina, The Object People, Siemens, Sun Microsystems, and Abbott Labs (for a complete listing visit our about us page at <http://www.refactory.com/our-company/our-clients>)

We also strongly believe that the best consultant is the one who is trying to put himself or herself out of a job.

Training

Our training utilizes hands-on teaching methods to provide the best possible course for your specific needs. Our practitioners develop courses to complement real world scenarios. The customization of our courses ensures that your team is taught the core principles necessary for their company projects and the overall direction of your company's development.

The Refactory offers unparalleled expertise in design patterns. Patterns distill recurring architectural, structural, and functional motifs employed by experienced designers when they construct their systems. Patterns have been proven to greatly facilitate the development process and give analysts, designers, and developers a common design vocabulary. Patterns play a central role in the iterative, incremental process. We provide a course originally designed by one of the authors of the Design Patterns book that covers how to use and implement design patterns into current development. This course also covers on how to evolve code into better designs by applying design patterns to your existing code-base.

The Principals and Associates of The Refactory, Inc. are the inventors of modern refactoring. We provide courses and mentoring on this topic. Refactoring is the process of taking code that may be reaching its limits of maintainability or extensibility, and evolving the code by applying design patterns and other known principles to a much better design that still works as well as it did before. There are automated tools that can assist with this. Our group has been the leader and instrumental in pushing this technology in industry including the only PhDs written on the subject matter and the development of the first automated tools for automatic refactoring.

Additionally we offer many other courses in areas such as Adaptive Object-Models, Frameworks, Pragmatic Test Driven Development and Agile Methods such as those outlined in eXtreme Programming (XP).

Firm's Qualifications

The Refactory, Inc. was founded in 1998 by internationally recognized experts on patterns, frameworks, refactoring, adaptive object-models, reuse, object-oriented development, architecture, analysis and design, testing, and agile practices. Our biographies and experiences can be found at www.refactory.com. Our staff offers a unique blend of practical experience and research acumen. Each principal brings doctoral level skills and many years of architecture, analysis, and design experience.

We have been working together for many years and we use similar methodologies, which will provide consistency for your group. We have extensive industry experience and like to apply our knowledge in practical pragmatic ways. We are familiar with Agile Practices, Object-Oriented Analysis and Design, Design Patterns, Frameworks, Refactoring, Reuse, Testing, and Industry Standard Best Practices for Object-Oriented Development. Our group has exceptional expertise in a number of areas that are particularly germane to your needs. Among these strengths are:

- Adaptive Architecture – Building Adaptive Systems, such as Adaptive Object Models, (www.adaptiveobjectmodel.com) makes it easier to allow users to add new rules and features to the system without new programming. We have pioneered research in this area and have assisted many industries including financial, medical, insurance and manufacturing in building these types of systems.

- Object Design and Patterns – Our organization boasts several leaders in this field. We have extensive experience on assisting with new people learning objects to advanced object-oriented techniques including design patterns. Our expertise in this area will translate into cleaner, smaller, better-factored, more comprehensible and maintainable code.
- Agile Development Practices – Our staff has been working with various Agile practices and other Lightweight Methodologies and their practical application for over fifteen years. This has exposed us to many new lightweight, nimble, agile style development processes that have emerged from this community. It has especially proven very useful for web-based applications.
- Architecture – The Refactory specializes in the development of software architecture through the use of good design, patterns of best practices, frameworks, refactoring, light-weight programming methodologies, testing, and building adaptable systems. Our Agile methods increase production speed, while maintaining adaptability and ensuring quality architecture.
- Cloud Computing/Web Services – We have been involved with the web since Mosaic emerged from NCSA in the early 90's. We have built a variety of object-oriented web-based applications, employing a wide range of different technologies. We also have extensive academic and commercial website design and implementation experience.
- Framework Development and Evolution – We have done groundbreaking work in the area of framework and object evolution. We have applied these insights as we have cultivated frameworks for a number of domains including Financial, Insurance, Medical and Manufacturing.
- Reuse – Reuse is often touted as one of the benefits of object-oriented technology. However, merely using object-oriented languages and tools will not make one's system reusable. It takes a gift for abstraction, patience, commitment, and experience to glean reusable classes, components and frameworks from the applications that spawn them. Our company knows how to make reuse work. We have built a number of successful object-oriented frameworks, and reused these frameworks to build new applications.
- Refactoring – Refactoring is the process of taking code that may be reaching its limits of maintainability or extensibility, and evolving the code by applying design patterns and other known principles to a much better design that still works as well as it did before. Our group has been the leader and instrumental in pushing this technology in industry including the only PhDs written on the subject matter and the development of the first automated tools for automatic refactoring.
- Security and Persistence – We have particular expertise in this area, and have incorporated our insights into a number of systems, including systems for Illinois Department of Public Health, Iron Mountain Fulfillment Services (financial and insurance) and Caterpillar Inc. Our work in this area has been published and validated as proven techniques in Industry.
- Pragmatic Testing – The Refactory, Inc. has been one of the primary proponents of Unit Testing and Test Driven development since its inception. We advocate a pragmatic approach. We believe testing should be a fundamental, integral part of software development process including design and programming. By fundamental, we mean that doing testing well leads to better designs and programs. By integral, we mean that it is part of the normal day-to-day life of the programmer, not a burdensome task added at the end of the real work.

Staffing

Experiences of the principles (John Brant, Brian Foote, Ralph Johnson, Don Roberts, and Joseph Yoder) can be found on our website at <http://www.refactory.com/our-company>. Joseph Yoder, Brian Foote, and Rebecca Wirfs-Brock are just a few of the experts available from The Refactory to provide your onsite training and mentoring. We can offer the above-mentioned people for training, mentoring, architectural, design, and documentation to meet any specific needs that you may have.

Joseph W. Yoder

Joseph W. Yoder is a founder and principal of The Refactory, Inc., a company focused on software architecture, design, implementation, consulting and mentoring on all facets of software development. Joseph is an international speaker and pattern author, long standing member of the ACM, and the President of The Hillside Group, a group dedicated to improving the quality of software development. Joseph specializes in Architecture, Analysis and Design, C#, Java, Smalltalk, Patterns, Agile Methods, Adaptable Systems, Refactoring, Reuse, and Frameworks. Joe is the author of many patterns, including being an author of the Big Ball of Mud pattern, which illuminates many fallacies in the approach to software architecture. Joe currently resides in Urbana, Illinois.

Joseph Yoder has taught, mentored, consulted, and managed various software projects for many years, including applying patterns, designing architecture, creating adaptive object models, utilizing agile methods, developing frameworks, refactoring existing systems, object oriented software development in Java, C#.NET, Smalltalk and C++, and web design and cloud services. Joe presents tutorials and talks, arranges workshops, and organizes leading technical conferences held throughout the world, including international conferences such as Agile, Agile Portugal, Encontro Ágil in Brazil, AOSD, CBSOft, JAOO, QCon, PLoP, AsianPLoP, SugarLoafPLoP in Brazil, OOPSLA, ECOOP, SATURN, and SPLASH. Joe thinks software is still too hard to change. He wants do something about this and believes that with good patterns and by putting the ability to change software into the hands of the people with the knowledge to change it seems to be on promising avenue to solve this problem.

Joe evolved from the Software Architecture and Patterns group at the University of Illinois. He has worked on various projects during his career that has incorporated many technologies. These range from stand-alone to client-server applications, web applications, web services, cloud computing, service oriented architecture, multi-tiered, various databases, object-oriented, frameworks, human-computer interaction, collaborative environments, and domain-specific visual-languages.

In addition, these projects have spanned many domains, including Medical Information Systems, Financial Systems, Ordering, Import, Invoicing, Print, Shipping, Warehouse Management, Manufacturing, Medical Examination, Statistical Analysis, Scenario Planning, Client-Server Relational Database System for keeping track of shared specifications in a multi-user environment, Telecommunications Billing System, and Business & Medical Decision Making. More recently, Joe has been teaching Agile Methods such as XP and TDD, Design Patterns, Object Design, Refactoring, and Testing in industrial settings. Additionally Joe has been consulting and mentoring people on these concepts, assisting teams in successfully building and deploying various systems. This recent work has also included working strongly in the .NET environment deploying frameworks and Domain-Specific Languages for some of our clients.

Ralph Johnson

Ralph Johnson is a professor at the Department of Computer Science at the University of Illinois. He mixes teaching there with industrial consulting. He is one of the leading pattern experts, and an expert on software reuse and object-oriented design.

Ralph Johnson has been working with objects since 1985. He has taught hundreds of people Smalltalk, some of whom are now widely recognized Smalltalk experts. His university course is taught over the internet.

He has worked on frameworks for compilers, operating systems, music synthesis, graphics editors, telephone billing systems, and insurance. He is recognized as an expert on object-oriented design and frameworks, having taught tutorials on framework design five times at OOPSLA, and consulted and taught framework design for many companies.

He is one of the four authors of the best selling "Design Patterns", and has been working with patterns for a long time, writing the first paper at OOPSLA on patterns (in 1992) and helping organize PLoP'94, the first conference on software patterns.

Ralph has been focusing on building success in software development by looking at ways to reduce the development cycle. This has led to his involvement with eXtreme Programming which he has recently sponsored as a four-week workshop at the University of Illinois for both Industry and Graduate level students.

Brian Foote

Brian Foote is an itinerant developer and rogue scholar who has been programming professionally since the late 1970s, when several years of immersion in abysmal scientific realtime code caused him to develop an interest in the question of why such smart people could write such lousy code. This question led him on a journey through the academy, where he studied objects, frameworks, patterns, and processes, and through industry, where he experienced the staggering gap between what we practice and what we preach about programming. His best known work, "Big Ball of Mud", examined this phenomenon a few years back. It still haunts him to this day.

Brian has over twenty years of professional software development experience. He has been working with Smalltalk and objects since 1985.

Brian has written numerous papers on Smalltalk, object-oriented design, software reuse, patterns, and software architecture. He used Smalltalk to prototype an extensive framework for scientific laboratory experimental control and data acquisition. He subsequently implemented this framework, OSIRIS, in C++ and sold it commercially. It is currently in use at several universities and research firms.

Brian was also involved in the development of the object-oriented enterprise frameworks developed at the Illinois Department of Public Health. His current research, on using objects to build better object-oriented languages, is being conducted in Smalltalk. Brian's other research interests include object-oriented programming, design, reuse, languages, frameworks, software architecture, patterns, reflection, metalevel architecture, and software evolution

Brain is the author of over two-dozen published patterns and has been working with patterns for a long time, writing his first pattern paper for the first PLoP conference in 1994, and chairing the PLoP '96, conference on software patterns.

Rebecca Wirfs-Brock

Rebecca Wirfs-Brock is an internationally recognized leader in the development of practical design methodologies. She invented the set of development practices known as responsibility-driven design. Among her widely used innovations are use case conversations and object role stereotypes. Rebecca has written, lectured, and presented tutorials on object design, adaptive architecture styles, and management topics. She is lead author of the best-selling book *Designing Object-Oriented Software* and *Object Design: Roles, Responsibilities and Collaborations*. She served four years as IEEE Software's editor of the Design column and is a past board member of the Agile Alliance.

Although best known as a software design guru she is also an innovator of techniques for simply expressing complex requirements and effectively developing and communicating software architecture. She frequently helps product engineering, IT, and startup organizations with technical strategies, frameworks and architecture as well as with effective teamwork and agile development practices. Rebecca's methodology contributions are based upon many years of hands-on software development experience. Most recently she has conducted architecture reviews of enterprise applications and product platforms, reviewed the design and implementation of a medical analytics framework, advised an engineering group on design and architecture for a large embedded application, and mentored teams on framework design, architecture, and managing incremental and iterative object-technology projects.

Her recent consulting projects include architecting an Operations Support Systems integration framework for a telco company; advising management, architects and lead designers of telephone switch control software; advising a large financial company on how to support flexibility and variation in their software system; technical oversight of a government application tracking youth offenders in the state of Oregon; architecture tradeoff analysis training and consulting for a state agency; agile feature-level use cases for a cell phone company; review of a product line architecture and design for the next generation of a market-leading web application; architecture review of integrated enterprise applications; and development of custom training and requirements workshops for several industries. She has developed and delivered enterprise application design, agile architecture, object design, UML, use case, and design-focused training and customized training to industrial clients.