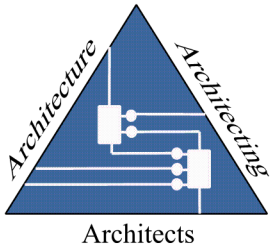
SOFTWARE ARCHITECTURE FUNDAMENTALS

from BREDEMEYER CONSULTING



http://www.bredemeyer.com Tel: (812) 335-1653

OVERVIEW

In this introductory class, we focus on the central concepts and key concerns of software architecture. It is organized around three major themes:

- *architecture*, answering the question: "what is software architecture?" This section deals with architecture definitions and distinctions, benefits, key concerns, and primary techniques, including an introduction to architectural patterns.
- *architecting*, that is the process of creating an architecture. We provide an overview of our Visual Architecting Process.
- *architects*, focusing on the role and responsibilities of the architect and positioning the architect role within the organization.

BENEFITS

More and more companies are recognizing that they need to build their organization's architecture competency, and do so quickly. Part of this process is introducing architecture concepts, and an appreciation for architecture and its role in software development, to a broad community of managers, architects, and developers. This is as true in product development as it is in IT, where application architecture is the foundation for competitive service offerings and effective internal systems.

This class is designed to help build a broad, shared appreciation for architecture and the role of architects, and the role of architecture in upstream (business) and downstream (development) processes. It helps build a common foundation of architecture terms and concepts, that will facilitate the work of architects, and the work of everyone who uses the architecture directly or indirectly.

We have considerable experience helping architects and their organizations transition to higher levels of architectural competency. This class integrates extensive real-world architecture experience, and builds on five years of teaching software architecture workshops.

COURSE OUTLINE

Software Architecture

In the software architecture section, we cover:

definitions and distinctions: we offer definitions of software architecture and distinguish it from other related kinds of architectures (enterprise, information, system, etc.).

benefits: we discuss the benefits of successful architectures, and the relationship between business strategy, architectural strategy and competitive advantage.

key concerns: we identify the focal concerns of architecture, including system decomposition and component specification, communication mechanisms, and ensuring system properties such as performance, consistency, data integrity, and scalability.

primary techniques: we outline the primary techniques used in architecting, including

- *architectural patterns*: we introduce structural patterns such as layers and client/server, and mechanisms such as brokers and bridges.
- *architecture modeling* using the Unified Modeling Language (UML).
- *architectural views*: structural views help document and communicate the architecture in terms of the components and their relationships. Behavioral views are useful in thinking through how the components interact to accomplish their assigned responsibilities and evaluating the impact of what-if scenarios on the architecture. Execution views help in evaluating physical distribution options and documenting decisions.

architecture, design and development: we discuss the relationship of architecture to design and implementation, and the various roles that are played in ensuring that the architecture is successful.

The Architecting Process

In the architecting section, we introduce our Visual Architecting process, and provide an overview of the steps involved in:

Init/Commit: how to gain the support of all levels of management affected by the architecture; how to create and lead a cohesive, creative architecture team

Architectural requirements: how to elicit and document functional (i.e., behavioral) requirements and non-functional requirements (i.e., system qualities).

System Structuring: how to use architectural modeling to decompose the system, evaluate architectural trade-offs, and document the system using different views.

Evaluation: how to assess the system in terms of the system requirements.

Deployment: how to assist the developer and project manager community in their use of the architecture to ensure its successful adoption and appropriate use.

The Role and Responsibilities of the Architect

This section describes the role of the architect, and discusses the responsibilities and associated skills and attitudes of the architect.

Format

This 1-day class relies mostly on lecture with class participation. We draw on stories from our experience, and large-group graphical facilitation, to help enliven and enrich this seminar format. Nonetheless, if you are looking for a more hands-on, and hence valuable, learning experience, we encourage you to attend our 4-day Software Architecture Workshop.

About the Instructors

The workshop leader, Dana Bredemeyer, has over 20 years experience architecting, designing and developing software systems, including 16 years with Hewlett-Packard. He developed Hewlett-Packard's internal Software Architecture Workshop, and continues to be its principal instructor. He has provided architecture consulting and training to architects, architecture teams and their management at the project, organization and business unit levels, and has helped teams develop software, firmware and system architectures for products, product families and information systems. Dana is currently writing a book on software architecture for Prentice-Hall.

Audience

This class is intended as an introduction to architecture for a broad audience, from managers, to technical leads, developers and test engineers.

Pricing

In-house: Pricing depends on number of students taking the class. Please contact us for details.
Open Enrollment: The cost for the 1-day class is US$500 per student. Lunch and snacks will be provided.

The price includes the workshop binder with color copies of the slides and notes.

Cancellation Policy

We will refund the registration fee less a 5% processing fee if cancellations are made more than 21 days in advance of the workshop start date. Cancellations made 21 days or less from the workshop start date are not refundable. However, student substitutions may be made any time prior to the start of the workshop.

We reserve the right to cancel or reschedule any workshop if enrollments are below the minimum necessary to cover costs and provide good class participation. This rarely happens, and we will keep registrants informed if enrollments are low enough to cause concern. If we are forced to cancel for any reason (such as major personal calamity including severe sudden illness or injury), our obligation will be limited to a full refund of class enrollment fees or transfer of the registration to a future workshop.

Venue

On-site: If you would like to hold this class in-house, please call us at (812) 335-1653 or email training@bredemeyer.com for pricing and schedule.

About Bredemeyer Consulting

Bredemeyer Consulting specializes in training and mentoring enterprise and software architects. We typically work with architecture teams, helping them to accelerate their creation or migration of an architecture. However, we do offer a limited number of Software Architecture and other workshops for open enrollment.

For more information, please

- see our web site: http://www.bredemeyer.com
- call 1 812 335-1653.