

## **Capability Cases: A Solution Envisioning Approach**

By Irene Polikoff, Robert Coyne, Ralph Hodgson

.....  
 Publisher: **Addison Wesley Professional**

Pub Date: **July 29, 2005**

ISBN: **0-321-20576-6**

Pages: **504**

[Table of Contents](#) | [Index](#)

### **Overview**

"In this innovative book, Irene Polikoff, Robert Coyne, and Ralph Hodgson have captured deep insights from many years of studying how to get across the 'Business-IT Gap.' Capability Cases are an invaluable tool for bridging from envisioning to design."

Steve Cook, software architect, Microsoft Corporation

"Capability Cases offers hope that mere mortals can rise above the daunting challenge of blending technology and process. Executives hoping to adopt high-performing business models will want to learn this step-by-step approach to designing and implementing IT-enabled organizational change."

Dr. Jeanne W. Ross, principal research scientist, MIT Center for Information

### **BRIDGE THE GAP BETWEEN BUSINESS VISION AND SOFTWARE SOLUTION**

Both IT organizations and business leaders recognize the urgent importance of aligning technology solutions with enterprise strategy. But they've struggled to do so... until now. Capability Cases represents a breakthrough: a powerful, systematic way to translate business vision into effective plans and system designs.

The authors' Solution Envisioning methodology addresses the crucial "front end" of software development, where you decide what to build and how you're going to build it. Using realistic examples, they walk you through exploring the capabilities you need, capturing business best practices, identifying optimal solutions, and crafting software systems that deliver them.

Solution Envisioning enables you to specify better solutions in less time, build systems that more fully reflect your true needs, and dramatically reduce risk and cost throughout the entire development process.

Understand Solution Envisioning from vision to plan in three phases:

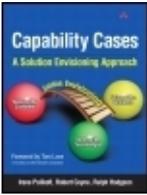
- Use Business Capability Exploration (BCE) to build a shared understanding of what is needed in a solution model your business situation using business forces, desired results, metrics, and scenarios
- Run Solution Capability Envisioning (SCE) workshops to facilitate decision-making, collaborative assessment, and consolidation of a solution concept use Capability Case Galleries to discover alternatives, gain context, promote stakeholder interaction, and stimulate creativity
- Move from concept to business case to a roadmap for realization with Software Capability Design (SCD)

Includes easy-to-use "best of class" templates for developing a business case, concept of operations, architectural decisions, and other key work products.

Whether you're an executive, architect, project manager, developer, change agent, or consultant, Capability Cases will help you bridge the gap between vision and solutions so you can finally get what you need from

information technology.

© Copyright Pearson Education. All rights reserved.



## **Capability Cases: A Solution Envisioning Approach**

By Irene Polikoff, Robert Coyne, Ralph Hodgson

.....  
 Publisher: **Addison Wesley Professional**

Pub Date: **July 29, 2005**

ISBN: **0-321-20576-6**

Pages: **504**

[Table of Contents](#) | [Index](#)

Copyright  
 Praise for Capability  
 Cases  
 About the Authors  
 Foreword  
 Preface  
 Acknowledgments  
 Introduction  
 TrailheadWhy This  
 Book?  
 What Is the Solution  
 Envisioning  
 Process?  
 What Are Capability  
 Cases?  
 Preview of  
 Important Terms for  
 Modeling the  
 Business Situation  
 Highlighting the  
 Main Ideas and  
 Threads of the Book  
 A Trail Map for the  
 Book: The Journey  
 Ahead  
 Part I. Solution  
 Envisioning: What?  
 and Why?  
 Chapter 1.  
 Technology  
 Innovation and the  
 Changing Business  
 Landscape  
 Chapter Preview  
 Understanding  
 Business  
 Challenges and IT  
 Requirements

The Changing  
 Role of IT  
 An Illustrative  
 Story of Solution  
 Envisioning  
 Trail Marker I:  
 Appreciation of  
 the Problem and  
 Trail Ahead  
 Chapter 2. Bridging  
 the Gap with  
 Solution  
 Envisioning  
 Chapter Preview  
 Solution  
 Envisioning with  
 Capability Cases  
 Process  
 Overview: From  
 Vision to Plan in  
 Three Phases  
 The Role and  
 Value of  
 Capability Cases  
 A Vocabulary  
 That Integrates  
 Envisioning and  
 Capability Cases  
 When Can  
 Solution  
 Envisioning Be  
 Used?  
 How Long Does  
 It Take?  
 Trail Marker II:  
 Envisioning  
 Solutions with  
 Capability Cases  
 Chapter 3.  
 Illustrative  
 Applications and  
 Galleries  
 Chapter Preview  
 Capability Case  
 Galleries  
 Case Study I:  
 Search  
 Technology  
 Workshop for the  
 State Government  
 Case Study II:  
 Solution  
 Envisioning at  
 NASA

Capability Cases  
as Solution  
Design Assets  
Trail Marker III:  
Exploring  
Capability Case  
Galleries

Chapter 4.  
Positioning Within  
the Solution  
Delivery Cycle  
Chapter Preview  
What Makes  
Solution  
Envisioning  
Different?  
Where Does It Fit  
in the Solution  
Delivery Process?  
Envisioning,  
Capability Cases,  
and the Iterative  
Nature of Design  
Trail Marker IV:  
Rationale for  
Solution  
Envisioning  
Outlined

Part II. Solution  
Envisioning  
ProcessDetailed Look  
Chapter 5. Business  
Capability  
ExplorationPhase I  
of Solution  
Envisioning  
Chapter Preview  
To Begin  
EnvisioningBusiness  
Capability  
Exploration  
(BCE)  
BCE Activities I:  
Establish  
Business  
Situation and  
Resources  
BCE Activities II:  
Identify Business  
Needs as Forces  
and Results  
BCE Activities  
III: Explore  
Possibilities

Informed by  
 Technology  
 BCE Activities  
 IV: Consolidate  
 Initial Solution  
 Vision for  
 Sharing  
 The Results of  
 BCE and  
 Readiness to  
 Move to the Next  
 Phase  
 Trail Marker V:  
 Business  
 Situation and  
 Solution Initiative  
 Understood  
 Chapter 6. Solution  
 Capability  
 EnvisioningPhase II  
 Of Solution  
 Envisioning  
 Chapter Preview  
 Solution  
 Capability  
 Envisioning  
 (SCE)Appraising  
 Solution Ideas  
 SCE Activities I:  
 Prepare for the  
 Solution  
 Envisioning  
 Workshop  
 SCE Activities II:  
 Conduct the  
 Solution  
 Envisioning  
 Workshop  
 SCE Activities  
 IIIPost-Workshop  
 Assessment and  
 Consolidation  
 Trail Marker VI:  
 Solution Concept  
 and Roadmap  
 Established  
 Chapter 7. Software  
 Capability  
 DesignPhase III of  
 Solution Envision  
 Chapter Preview  
 Software  
 Capability Design  
 (SCD)From

Concept to  
 Realization  
 SCD Activities I:  
 Select  
 Implementation  
 Technologies  
 SCD Activities II:  
 Develop Software  
 Capability  
 Architecture  
 SCD Activities  
 III Develop  
 Business Case  
 and Solution  
 Roadmap  
 The Results of  
 Software  
 Capability Design  
 Moving from  
 Solution  
 Envisioning to  
 Solution  
 Realization  
 Trail Marker VII:  
 Business Case  
 and Roadmap for  
 Implementation  
 Part III. Customizing  
 and Using Solution  
 Envisioning  
 Chapter 8. Solution  
 Envisioning in  
 Different Situations  
 Chapter Preview  
 Solution  
 Envisioning as a  
 Customizable  
 Framework  
 Focusing the  
 Solution  
 Envisioning  
 Workshop  
 Different  
 Situations Different  
 Approaches  
 The  
 Complementary  
 Role of Solution  
 Envisioning  
 Trail Marker VIII:  
 Solution  
 Envisioning  
 Tailored to  
 Situation

Chapter 9.  
 ConclusionLessons  
 Learned and  
 Looking Ahead with  
 Envisioning

Reflections on  
 Solution  
 Envisioning  
 Plans for  
 Evolving Solution  
 Envisioning  
 Fostering  
 Adoption and  
 Practice of a New  
 Approach  
 End of Trail:  
 Solution  
 Envisioning  
 Endnotes

Appendix A.  
 History and Design  
 of Solution

Envisioning  
 The X-Model and  
 Object-Oriented  
 Development  
 Design by  
 Analogy and the  
 Architecture  
 Handbook  
 OOPSLA  
 Workshops(19962000)  
 Other Seminal  
 Influences  
 Solution  
 Envisioning and  
 Creativity  
 Why Metaphors  
 Are Powerful  
 Capability Cases  
 and Patterns  
 The Importance  
 of the "Right"  
 Deliverables

Appendix B.  
 Additional Techniques

Creativity  
 Techniques  
 Decision Support  
 Techniques  
 Conducting  
 Envisioning  
 Workshops as  
 "Theater Play"

Appendix C. Software  
 Capability Design  
 Implementation  
 Alternatives  
   SCD for Custom  
   Applications  
   SCD in  
   EDOC-Based  
   Component  
   Development  
   SCD and Package  
   Implementation  
   Conclusion  
   Endnotes  
 Appendix D. Solution  
 Envisioning Activities  
 and  
 WorkproductsSummary  
 Table for Each Phase  
   The Activities and  
   Workproducts of  
   Phase IBusiness  
   Capability  
   Exploration  
   The Activities and  
   Workproducts of  
   Phase IISolution  
   Capability  
   Envisioning  
   The Activities and  
   Workproducts of  
   Phase IIISoftware  
   Capability Design  
 List of Figures  
 List of Tables  
 List of Techniques  
 Glossary  
 References  
 DRAFT  
 MANUSCRIPT  
 Color Insert  
 Index

## Copyright

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and the publisher was aware of a trademark claim, the designations have been printed with initial capital letters or in all capitals.

The authors and publisher have taken care in the preparation of this book, but make no expressed or implied warranty of any kind and assume no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or

programs contained herein.

The publisher offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales, which may include electronic versions and/or custom covers and content particular to your business, training goals, marketing focus, and branding interests. For more information, please contact:

U. S. Corporate and Government Sales  
(800) 382-3419  
[corpsales@pearsontechgroup.com](mailto:corpsales@pearsontechgroup.com)

For sales outside the U. S., please contact:

International Sales  
[international@pearsoned.com](mailto:international@pearsoned.com)

Library of Congress Catalog Number: 2005923980

Copyright © 2006 Pearson Education, Inc.

All rights reserved. Printed in the United States of America. This publication is protected by copyright, and permission must be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or likewise. For information regarding permissions, write to:

Pearson Education, Inc.  
Rights and Contracts Department  
One Lake Street  
Upper Saddle River, NJ 07458

Text printed in the United States on recycled paper at R. R. Donnelley in Crawfordsville, Indiana.

First printing, July 2005

## Dedication

This Book is dedicated to "embattled envisioners" and "change agents" everywhere. May you follow and guide others on interesting trails of possibilities...

## Praise for Capability Cases

"I've known the authors of this book for some years, and have used the Solution Envisioning through Capability Cases approach on quite a few major IT engagements for fortune 500 client companies. It's an outstanding technique for ensuring real consensus between stakeholders as to the scope and priorities of an IT project. I heartily recommend this book to anyone involved in creating technology-based solutions to business problems."

Clive Gee, Ph.D., senior solution architect, IBM Enterprise Integration Services

"Capability Cases clarifies the mysterious process whereby business needs inform, and are informed by, technology. The formalism of Capability Cases provides a way for real-world

successes to be abstracted into best practices. The guiding principle is 'clarity,' and the book will be clarifying for any reader concerned with these issues."

Alexander Morgan, principal research scientist, Manufacturing Systems Research Laboratory, GM R&D Center

"Solution envisioning is critical to the 'learning your way to innovation' that we need for breakthrough performance in e-government public services and the build-out of its target architecture."

Brand Niemann, co-chair, Federal Semantic Interoperability Community of Practice (SICoP) and Office of Environmental Information, U.S. Environmental Protection Agency

"The methodologies described in Capability Cases begin to define a structured approach that connects the creative envisioning of potential solutions with a pragmatic process for defining and implementing techniques and technologies. These methods are particularly relevant in the diverse and disparate arena of biomedical sciences and application of bioinformatics in the healthcare and basic biology."

Dave Parrish, Immune Tolerance Network

"Capability Cases offers hope that mere mortals can rise above the daunting challenge of blending technology and process. Executives hoping to adopt high-performing business models will want to learn this step-by-step approach to designing and implementing IT-enabled organizational change."

Dr. Jeanne W. Ross, principal research scientist, MIT Center for Information Systems Research

"In this innovative book, Irene Polikoff, Robert Coyne, and Ralph Hodgson have captured deep insights from many years of studying how to get across the 'business/IT gap.' Capability Cases are an invaluable tool for bridging from envisioning to design."

Steve Cook, software architect, Microsoft Corporation

"This book is a welcome addition to the discussion of the perennial problem of bridging the business/IT gap. The authors bring years of practical experience in addressing such issues as confusing proposed solution attributes with business needs and requirements, balancing the needs and expectations of different classes of stakeholders, and familiar syndromes such as the package will do it and the demo becomes the solution. These issues all too often lead to project failure, as cited in an AMR study of CRM projects, of which only 16% delivered actual business value.

The authors prescribe an approach based on Solution Envisioning and Capability Cases. They provide a compelling rationale for their suggestions, clear instruction for practitioners, and guidance for the incorporation of these methods into existing practice. They make use of such techniques as stories and an emphasis on clear definition of vocabulary to help balance the voice of the user and the voice of the technologist. The book contains galleries of actual Capability Cases and tips for running effective workshops, including techniques for interaction and for creating an effective environment for collaboration. This approach incorporates both standard and innovative methods and tools for recognizing business situations, developing concept requirements, and the application of technology capabilities to realize desired business capabilities.

Based on real-life examples from commercial and public enterprises, and sprinkled with war stories and entertaining cartoons, this book delivers methodical wisdom... and more."

Doug McDavid, Almaden Services Research, IBM Academy of Technology  
([mcdavid@us.ibm.com](mailto:mcdavid@us.ibm.com))

"Capability Cases fills in a critical void. Many agile approaches focus on micro-requirements stories, features, backlog items assuming that the user community has conducted adequate business or product analysis. We know requirements change frequently and therefore don't want to specify reams of detailed specifications, but what artifacts should we use to guide projects in that intermediate zone between product vision and iteration specifications? The answer: Capability Cases. Capability Cases are one answer to the 'just enough, but not too much' front-end work that's required to link technology to business value."

Jim Highsmith, senior vice-president, Agile Software Development and Project Management Practice, Cutter Consortium

## About the Authors

Irene Polikoff, partner and co-founder of TopQuadrant, Inc., has more than 15 years of IT experience. Her specialties include technology strategy, IT management, system integration, process reengineering, systems assessment, and technology selection.

Dr. Robert Coyne, executive partner at TopQuadrant, has more than 20 years of experience in the full lifecycle of business development, product design, software development, production, and delivery. He brings integrated knowledge and expertise from business, consulting, academia, and research to solution development processes, methods, and practice.

Ralph Hodgson, partner and co-founder of TopQuadrant, has been a thought leader in the IT industry for some 25 years. He has expertise in enterprise systems design, software engineering, IT consulting, object and ontology modeling, Semantic Technology, logic programming, and methodology development.

## Foreword

Professor Fred Brooks refers to "thinkers," "do-ers," and "thinker-doers." This book is written by some of the most energetic "thinker-doers" in the business. They are amazingly well-versed in current academic research and leading edge technology. They also get their hands dirty and deliver specifications, designs, and operational software for their customers.

The book adds an important term to our already large catalog of terms that we system builders routinely use: Capability Cases. The essence of the book is that just as we need reusable components to improve our productivity while developing systems, we also need a reusable library of Capability Cases to help explain to our customers what is possible and to comfort them by knowing that "what already exists cannot be impossible."

An envisioned solution achieved through some suitable activities, such as those defined as the "Solution Envisioning" aspect of the book, then is a collection of Capability Cases. Just like reusable patterns, Capability Cases describe something which already exists from a user's perspective, not a developer's perspective. By combining a collection of Capability Cases a set of users and stakeholders can get a more precise notion of

what could be built and what business value that particular solution would provide.

Why is this book important? It is important because far too many business people in the world are simply catatonic at the thought of being responsible for "delivering a custom application." It need not be so. A custom application can be described using a set of reusable Capability Cases; those Capability Cases can be implemented with a set of reusable frameworks or components; and everything is held together with a modest amount of "glue code," which is not very risky to implement. With such a strategy businesses can finally get the applications and systems that they want instead of the systems they are being sold which always require the business to change to accommodate the application or system.

Years ago I had the privilege of managing a collection of remarkable "thinker-doers" at ITT Corporation. We were asked to develop a 10 year projection of software development technologies. We constrained ourselves in our projections by only considering capabilities which already existed, even in a high technology research group. Surprisingly we did a great job of predicting where hardware technology was going and we were a decade optimistic in our view of software technology. So constraining ourselves to think about solutions composed only of Capability Cases is not very restrictive!

If we restrict our thinking about new systems to existing Capability Cases, does this ensure that we will not have failed projects? Of course not. The project may still fail because we attempt to do too much too quickly or with too few developers or with inexperienced developers.

What if we're a startup company and really want to develop something that is substantially different from anything that has ever been built before? It certainly can be done, but some cautions apply. One or two major innovations are all that a single project can accommodate. I have seen companies fail when they try to develop too many fundamental innovations in parallel. But there is nothing wrong with a series of innovations happening in sequence. Companies like Apple and NeXT have demonstrated their ability to do just this. Microsoft has demonstrated spectacular prowess in taking innovations developed elsewhere and commercializing them with or without the cooperation of the original developer!

Ralph (one of the authors of this book who has been working on software projects as long as I have) and I remember when companies would spend millions of dollars writing meaningless specification documents filled with conflicting vague notions of what some set of stakeholders thought they wanted in a new system. Designers and developers would ponder for days and weeks and sometimes even months trying to figure out what on earth the customer really wanted. Until the solution is completely described in a form that designers and developers can unambiguously understand what It<sup>[1]</sup> is, no one can provide an accurate estimate of the time and effort to construct It (using whatever technology).

[1] It is a pronoun and refers to the envisioned solution; IT is an acronym for Information Technology.

Capability Cases, in conjunction with the Solution Envisioning process, provide for a more precise way to define It. With such precision, IT organizations can finally start delivering the business value that their stakeholders demand but rarely have received. Irene, Ralph, and Robert have made a fundamental contribution to software engineering. I look forward to seeing a virtual bookshelf full of books describing available Capability Cases. It will happen soon, I predict.

Next will come the tools to assemble Capability Cases with their associated object-oriented components. This is not an easy project and would be quite hard to express using existing Capability Cases. So be prepared to wait a decade or so before our often described building blocks solution can be assembled from Capability Cases.

Tom Love, co-founder and CEO, Shoulders Corporation

## Preface

Capability.<sup>[1]</sup> It's a promising word. It sounds, well, capable, and it alludes to the delivery of something. We all use capabilities every day in the goods and services by which we construct our lives. We like to be capable; we respect and value capabilities in others. As individuals, we carefully evaluate and compare the capabilities in the products and services we purchase in everything from cars and houses to palm pilots, digital cameras, and mobile phone services. We look for and expect capabilities in companies we choose to deal with and in the companies we own or choose to be a part of.

<sup>[1]</sup> The quality of being capable; capacity or ability needed to do something. (plural) Qualities that may be used or developed; potentialities.

As businesses, we also need to purchase or build, deploy and use, offer or consume capabilities all the time regularly and continuously. It is requisite that we absorb, embrace, employ, and evolve new capabilities with ever greater speed and agility simply to remain effective and cost competitive. But, knowing about and successfully deploying the right new capabilities has always been challenging to businesses. In particular, software and technology capabilities are sometimes as elusive, amorphous, and difficult to pin down as they are valuable and essential to the core operations and processes of most modern businesses and organizations.

Most business capabilities today are a combination of human capital, fixed assets (such as facilities), processes, and technology. Forging successful combinations requires proficient orchestration of the interplay between business and technology. In this era of technology explosion (ranging from Web Services to new Platforms for Collaboration and Personalization to Semantic Technologies and Agents), the possibilities are as truly mind-boggling as the challenges of effective action. In nearly every industry and business area, technology capabilities are playing an ever more prominent role and some business capabilities are becoming identical with technology capabilities.

We are entering a world where the network is the platform and a large amount of business-enabling capabilities will be made possible by assembling commodity software that implements standardized protocols. Indeed, technologies are creating deeper changes in businesses invoking a disruptive new order that challenges all the rules of business. Business capabilities made possible by technology demolish traditional barriers of geography, law, organization, and time while simultaneously raising new bars for success in terms of connectedness, convenience, quality, and performance.<sup>[2]</sup>

<sup>[2]</sup> We are indebted to Peter Stecher of IBM, EMEA, for insights into this characterization of the revolutionary nature of new technologies in terms of the deeper changes they bring to doing business.

For instance, capabilities introduced by technology have changed user expectations. Business has become more personal, and customers have (much) more control. Businesses have rapidly become more connected and accessible, and in many cases instantly available all the time. As a result, business has paradoxically become both simpler and more complex. Technologies can simplify transactions, but they also engender more complex transactions and more sophisticated use of data and knowledge. New capabilities create challenges to business models by regularly shifting the value proposition. Many things that were previously sold are given away for free. Business value, to be earned by truly serving customers, has shifted to somewhere else in the value chain.

How hard it is to identify, design, and deploy the right capabilities? Capability, per se, doesn't usually conjure up any nuances or overtones of failure (yet). But, as we all know, as soon as we attempt to acquire and make use of capabilities, there is suddenly a pressing demand for a blend of vision, planning, and know-how to take advantage of and make capabilities actual or realized through their successful application. Further complicating the situation, there is an increasing multiplicity of sources from which to realize (develop, acquire, compose) capabilities. Geoffrey Moore, industry consultant and best-selling author, puts it this

way:<sup>[3]</sup>

<sup>[3]</sup> From Gartner Application Integration & Web Services Summit 2005 brochure, highlighting Moore's keynote talk, "The Key to Sustained Leadership: Separating the Core from the Context," where he "shares his latest insights into the software industry challenges and opportunities."

"With service-oriented architecture, integration, composition of applications, open source software, vendor consolidation and the tension between building, buying and outsourcing software solutions companies are under pressure to change too much and too fast. Risks seem unavoidable. There is not enough budget or time to cover all the bases. The leading enterprises will identify the core of their business and invest in safety and agility of the IT behind this core first."

Unless we have knowledge of capabilities, the vision to know their worth, and the experience and discipline to apply them, we may get lost somewhere in a "sea of potentials." In our personal lives, in assessing where true capabilities are found, whether they are real, sound, and worth having, we typically use many sources of information, processes, and tools for evaluation facilities such as Consumer Reports<sup>[4]</sup> for consumer products or recommendations from friends or people who have experience with the capability. This book is an attempt to aid individuals and companies in improving technology-enabled business capabilities by responding to their needs for

<sup>[4]</sup> A source that has its own 'capabilities' a trusted source of information, knowledge, evaluative mechanisms, and the experience to make useful judgments about the relative promise and value of other capabilities.

- Understanding what capabilities exist what is out there
- Facilitating the interplay between business and technical ideas
- Understanding the significance and implications of capabilities and the constraints and costs of employing them
- Establishing effective processes for acquiring, adopting, and applying capabilities

We introduce the Solution Envisioning process to help people envision together the needs and possibilities of business. The process is supported by the Capability Case, an aid to understanding and evaluating capabilities. In the everyday sense of the words, the Capability Case is intended to build a case for a capability by illustrating its potential and value. We offer Solution Envisioning with Capability Cases (the approach this book introduces) as a start toward a new kind of essential, comprehensive approach for appreciating and making intelligent decisions about the available technology capabilities that will be powering businesses for some time to come.

## Acknowledgments

Many people inspired, encouraged, and supported us in a long journey that preceded writing this book. We trace the beginnings of System Envisioning ideas back to 1991 and Ralph Hodgson's work on the X-Model. The ideas took shape and evolved with the help of many participants of the ACM OOPSLA System Envisioning workshops from 1996 to 2000.

We acknowledge the beneficial experiences and ideas that flowed in the OOPSLA System Envisioning years. Special thanks to Martine Devos who co-led some of the OOPSLA workshops. Martine has always been an inspiration to us as a facilitator and "seasoned envisioner." Her ConceptCafe technique is one of the techniques used in our method and described in this book. Our appreciation of the OOPSLA workshops would not be complete without mentioning the contributions of Alistair Cockburn, Branko Peteh, Brian Foote, Bruce

Anderson, Carl Ballard, Charles B. Harvey, Chuck Matthews, Clive S. Gee, Dana Bredemeyer, Dave Sapp, Dave Thomas, David Ing, Deborah Leishman, Doug McDavid, Gerald Zinke, Ian Graham, Ian Simmonds, Jack Ring, Jim Salmons, John Daniels, Kal Ruberg, Kent Beck, Lorette Cameron, Mary Lynn Thomas, Marilyn Bates, Mark Simos, Michael Beedle, Mike Karchov, Petri Pulli, Ruth Milan, Steve Johnson, Tom Bridge, Trygve Reenskaug, and Ward Cunningham.

Our colleague, Dean Allemang, helped us at the start of the writing, and we extend thanks for the many hours over the last two years he devoted to reviewing, discussing, and helping to fine-tune various aspects of the book. We especially acknowledge Dean for his affinity and aptitude for doing the "Envisioning Thing" for semantic technology in our TopQuadrant customer projects. Some people are born to envision. In our view, Dean should be counted among them.

We doubt that any book author ever had a reviewer more conscientious than Larry Levine, who carefully read almost every chapter and provided us with copious notes and valuable feedback. Tom Love, Chris Newlon, Peter Stecher, and Alistair Cockburn provided many insightful observations making it possible for us to sharpen the writing and significantly improve the organization of the book. We also thank Mills Davis for helping shape the introductions for the three parts of the book. And we thank Sidney Bailin for the insights that became part of the storytelling technique.

This book would not be possible without many consulting clients who let us practice Solution Envisioning and learn from them. Many have reviewed the book, and allowed us to use workshop photographs and project examples. Our sincere thanks go to Alec Morgan, Bob Dompe, Christopher Pardy, Con Kenney, Frehiwot Fisseha, Gail Hodge, Gauri Salokhe, Hansen Wu, Harold P. Frisch, Irene Onyancha, Jane Riddle, Jim Cockrell, Johannes Keizer, John Zimmerman, Kafkas Caprazli, Larry Schmidt, Margherita Sini, Paul Keller, Randy Hoffman, Rusty Yates, Stefka Kaloyanova, and Stephen Waterbury.

We feature several Capability Cases and solution stories in the book. We appreciate the cooperation and assistance from the companies and individuals who gave their permission to include these in the book. These include the staff at the iMarkup Solutions, Inxight Software, Inc., Land's End, Inc., London's Transport Museum, Ontoprise and Right Now Technologies, Andrew Bradley of the Hive Group, Dallas Noyes of Cogito, Inc., Professor Jim Hendler of the MindLab and the University of Maryland, John Domingue of the UK Open University, Paul Kogut of Lockheed Martin, Peter Clark of Boeing, Dr. Thomas Vögele of GEIN, and Tony Frazier of iPhrase, Inc.

Artist and visual facilitator Peter Durand of Alphachimp drew all the excellent cartoons you see in the book. Graphic artist James Huckenhahler took many sketchy diagrams we provided and turned them into professional graphics. Finally, our gratitude goes to the staff at Addison-Wesley to Michael E. Hendrickson for making it possible to start the project and to Paul Petralia, our editor throughout the rest of the process, for helping to make the book a reality.

Finally, we would especially like to thank our families and friends for their understanding and support during the extended time and effort we needed for writing. We could not have envisioned nor realized this book without you.

[\[View full size image\]](#)

Preamble: Tales of the Quest for a Business Solution

[\[View full size image\]](#)

The Need to DO SOMETHING

LAUNCHING A SOLUTION INITIATIVE (\$\$)

[\[View full size image\]](#)

FIELDING A SOLUTION

[\[View full size image\]](#)

FIELDING A SOLUTION

[\[View full size image\]](#)

## FIELDING A SOLUTION

ACHIEVING (DESIRED?) RESULTS (some possible outcomes...)

[\[View full size image\]](#)

[\[View full size image\]](#)

## Introduction

"The key to every man is his thought. Sturdy and defying though he look, he has a helm which he obeys, which is the idea after which all his facts are classified.

He can only be reformed by showing him a new idea which commands his own."

Ralph Waldo Emerson

## TrailheadWhy This Book?

In this book, we focus on the interplay between business needs and technology capabilities. Our belief in the following premises has motivated its topic and contents:

- Clarity of the business problem is central to getting results from technology projects. Strong functional fit of business solutions happens when the decision processes are value-driven.
- Innovative solutions emerge from understanding needs and appreciating technology capabilities. Because technology options continue to expand, this is not a simple task.
- Creativity rarely happens on its own. For ideas to flow, they need a supportive environment and skillful techniques to bring real people together to dialogue differently and more productively.
- Capabilities serve as a unifying construct across the lifecycle. A capability-based approach can span from the design of new business capabilities and the envisioning of solution capabilities to the specification of software capabilities.

We present two innovations that work together to better connect business challenges to technology enablers for successfully delivering value:

- Solution Envisioning is a business, value-driven approach to designing a system that uses Capability Cases and scenarios to foster innovation and to validate and increase confidence in the solution.