

Compliments of  blueprint

Requirements Definition & Management

FOR
DUMMIES[®]
A Wiley Brand

Learn:

- The critical role of the Business Analyst
- How Business Analysts are transforming business
- How Business Analysts are changing the way requirements are defined and managed
- What Business Analysts need to succeed
- Ten tips to improve Business Analyst productivity

Robert D. Schneider
Tony Higgins
Keith Barrett



***Requirements
Definition &
Management***
FOR
DUMMIES®
A Wiley Brand

***Requirements
Definition &
Management***
FOR
DUMMIES®
A Wiley Brand

**by Robert D. Schneider,
Tony Higgins and
Keith Barrett**

FOR
DUMMIES®
A Wiley Brand

Requirements Definition & Management For Dummies®

Published by
John Wiley & Sons Canada, Ltd.
6045 Freemont Blvd.
Mississauga, ON L5R 4J3
www.wiley.com

Copyright © 2013 by John Wiley & Sons Canada, Ltd.

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons Canada, Ltd., 6045 Freemont Blvd., Mississauga, ON L5R 4J3, or online at www.wiley.com/go/permissions. For authorization to photocopy items for corporate, personal, or educational use, please contact in writing The Canadian Copyright Licensing Agency (Access Copyright). For more information, visit www.accesscopyright.ca or call toll free, 1-800-893-5777.

Trademarks: Wiley, the Wiley logo, For Dummies, the Dummies Man logo, A Reference for the Rest of Us!, The Dummies Way, Dummies Daily, The Fun and Easy Way, Dummies.com, Making Everything Easier, and related trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates in the United States and other countries, and may not be used without written permission. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc. is not associated with any product or vendor mentioned in this book.

LIMIT OF LIABILITY/DISCLAIMER OF WARRANTY: THE PUBLISHER AND THE AUTHOR MAKE NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE ACCURACY OR COMPLETENESS OF THE CONTENTS OF THIS WORK AND SPECIFICALLY DISCLAIM ALL WARRANTIES, INCLUDING WITHOUT LIMITATION WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE. NO WARRANTY MAY BE CREATED OR EXTENDED BY SALES OR PROMOTIONAL MATERIALS. THE ADVICE AND STRATEGIES CONTAINED HEREIN MAY NOT BE SUITABLE FOR EVERY SITUATION. THIS WORK IS SOLD WITH THE UNDERSTANDING THAT THE PUBLISHER IS NOT ENGAGED IN RENDERING LEGAL, ACCOUNTING, OR OTHER PROFESSIONAL SERVICES. IF PROFESSIONAL ASSISTANCE IS REQUIRED, THE SERVICES OF A COMPETENT PROFESSIONAL PERSON SHOULD BE SOUGHT. NEITHER THE PUBLISHER NOR THE AUTHOR SHALL BE LIABLE FOR DAMAGES ARISING HEREFROM. THE FACT THAT AN ORGANIZATION OR WEBSITE IS REFERRED TO IN THIS WORK AS A CITATION AND/OR A POTENTIAL SOURCE OF FURTHER INFORMATION DOES NOT MEAN THAT THE AUTHOR OR THE PUBLISHER ENDORSES THE INFORMATION THE ORGANIZATION OR WEBSITE MAY PROVIDE OR RECOMMENDATIONS IT MAY MAKE. FURTHER, READERS SHOULD BE AWARE THAT INTERNET WEBSITES LISTED IN THIS WORK MAY HAVE CHANGED OR DISAPPEARED BETWEEN WHEN THIS WORK WAS WRITTEN AND WHEN IT IS READ.

For general information on our other products and services, or how to create a custom For Dummies book for your business or organization, please contact our Business Development Department at 877-409-4177, contact info@dummies.biz, or visit www.wiley.com/go/custompub. For information about licensing the For Dummies brand for products or services, contact BrandedRights&Licenses@Wiley.com.

For technical support, please visit www.wiley.com/techsupport.

Wiley publishes in a variety of print and electronic formats and by print-on-demand. Some material included with standard print versions of this book may not be included in e-books or in print-on-demand. If this book refers to media such as a CD or DVD that is not included in the version you purchased, you may download this material at <http://booksupport.wiley.com>. For more information about Wiley products, visit www.wiley.com.

ISBN 978-1-118-64113-2 (pbk); ISBN 978-1-118-64114-9 (ebk)

Manufactured in the United States

1 2 3 4 5 DP 17 16 15 14 13

About the Authors

Robert D. Schneider is a Silicon Valley–based technology consultant and author. He has provided database optimization, distributed computing, and other technical expertise to a wide variety of enterprises in the financial, technology, and government sectors. He has written six books and numerous articles on database technology and other complex topics such as cloud computing, big data, developing and testing high quality software, and service oriented architecture (SOA). He is a frequent organizer and presenter at technology industry events worldwide. Robert blogs at rdschneider.com.

Tony Higgins is the vice president of Product Management at Blueprint Software Systems. He has over 28 years' experience in the software industry in a variety of technical and business roles in the development of space and defense systems, IT systems, and commercial software products. For the last nine years Tony has focused specifically on solutions for software requirements definition and management and the business analyst role.

Keith Barrett is a senior systems engineer at Blueprint Software Systems and co-host of the Requirements Unplugged podcast series from www.requirements.net. Keith is a 22-year veteran of the software industry. Prior to joining Blueprint Software Systems, Keith served in the roles of business development manager, director of professional services, development manager, and practice manager for various software organizations. Keith has also been a project manager, developer, data modeler and business analyst in large financial and telecommunications corporations. Keith's background in software development, reuse, and requirements uniquely positions him as a leading industry advocate for business analysis best practices. His work on requirements and component-based design has been featured in *Business Analyst Times*, *Application Development Trends*, *ComputerWorld*, and *Object Magazine* and he has spoken at numerous conferences globally.

Publisher's Acknowledgments

We're proud of this book; please send us your comments at <http://dummies.custhelp.com>. For other comments, please contact our Customer Care Department within the U.S. at 877-762-2974, outside the U.S. at 317-572-3993, or fax 317-572-4002.

Some of the people who helped bring this book to market include the following:

Acquisitions and Editorial

Associate Acquisitions Editor:

Anam Ahmed

Production Editor: Lindsay Humphreys

Copy Editor: Jeremy Hanson-Finger

Editorial Assistant: Kathy Deady

Reviewer: Jennifer Bentley

Composition Services

Sr. Project Coordinator: Kristie Rees

Layout and Graphics:

Joyce Haughey, Andrea Hornberger

Special Art: Tony Higgins

Proofreaders: John Greenough,
Jessica Kramer

John Wiley & Sons Canada, Ltd.

Deborah Barton, Vice President and Director of Operations

Jennifer Smith, Vice-President and Publisher, Professional Development

Alison Maclean, Managing Editor, Professional Development

Publishing and Editorial for Consumer Dummies

Kathleen Nebenhau, Vice President and Executive Publisher

David Palmer, Associate Publisher

Kristin Ferguson-Wagstaffe, Product Development Director

Publishing for Technology Dummies

Andy Cummings, Vice President and Publisher

Composition Services

Debbie Stailey, Director of Composition Services

Table of Contents

Introduction	1
Foolish Assumptions	1
How This Book Is Organized	2
Chapter 1: Meeting the Business Analyst	2
Chapter 2: Conquering the Challenges of Business Analysis	2
Chapter 3: Building a Better Business Analysis Experience	2
Chapter 4: Introducing Blueprint.....	3
Chapter 5: Ten Tips to Improve Business Analyst Productivity.....	3
Icons Used in This Book.....	3
Where to Go from Here	4
Chapter 1: Meeting the Business Analyst	5
Boldly Exploring the Dynamic Business Environment.....	6
Engaging the Stakeholders.....	7
Analyzing the (Business) Analyst	9
Education.....	9
Work background	9
Place in the organization	10
Responsibilities.....	10
Career trajectory	11
Software development environments	12
Imagining the Future.....	13
Chapter 2: Conquering the Challenges of Business Analysis	15
Coping with Diverse Software Development Methodologies	16
Coordinating teams	17
Sprinting to milestones	18
Clearing the backlog.....	18
Managing Convoluted Business Processes	19
Overcoming Inadequate Tooling.....	20
Writing when you could be drawing	21
Typing when you don't need to	22
Doing clerical work that's not your job	22



Conquering Cumbersome Review Cycles 22
Exploring the Economic Risks of Inefficient
Business Analysis..... 24

**Chapter 3: Building a Better Business Analysis
Experience27**

Taking Strategies from Other Industries..... 27
Getting Visual 28
Getting Social..... 31
Getting Going..... 34
 Reengineering processes 34
 Rolling out software 34
Forecasting the Future 35

Chapter 4: Introducing Blueprint37

Highlighting the History of Blueprint 37
Sorting through Blueprint Solutions 38
 Authoring requirements 38
 Validating requirements 39
 Managing requirements 40
 Collaborating with team members 41
 Integrating with other technologies 42
Watching Blueprint at Work..... 43
 Banking industry solutions..... 43
 Legal industry solutions 44
 Health industry solutions 44
 Legacy modernization solutions..... 44

**Chapter 5: Ten Tips to Improve Business Analyst
Productivity45**

Leveraging Visual Models 45
Expressing Your Requirements..... 46
Setting the Stage for Effective Collaboration..... 47
Analyzing Stakeholders 47
Getting the Right Tool for the Job 48
Avoiding Being a Slave to the Document 49
Creating a Traceability Strategy 50
Embracing the Rationale..... 50
Planning Your Integration with Development
and Testing 51
Starting Small and Evolving Over Time..... 51

Introduction

Welcome to *Requirements Definition & Management For Dummies!* Enterprises of all shapes and sizes have become increasingly reliant on software to manage and optimize every phaser — er, phase — of their operations. In fact, software often provides a competitive edge that translates into enhanced profitability for the company.

Project teams that design and deliver business applications serve the needs of a broad range of stakeholders, including the business, user communities, and IT itself. Within these projects the business analyst plays a critical role as the main driver of requirements definition and management. Yet the position of business analyst is relatively new and many people still misunderstand what it entails.

In this book, we introduce you to the position of the business analyst, explain what they do, and describe their typical background. We also describe some of the major challenges facing these talented professionals, as well as how employing a combination of targeted technology and best practices can make business analysts more effective. And if business analysts are more effective, the entire organization can realize the full benefits of its software assets.

Foolish Assumptions

Although it's never a good idea to take anything for granted, we do have some beliefs about our readers. First, we expect that you work in either IT or the line of business, and that your job somehow involves specifying, designing, developing, or testing business software. We also assume that you're interested in improving the way your organization delivers these critical systems.

How This Book Is Organized

We designed the five chapters of this book to give you a comprehensive understanding of requirements definition and management, business analysts and the challenges and opportunities that they face, and how modern technology can come to the rescue.

Chapter 1: Meeting the Business Analyst

This chapter illustrates several business realities that have led enterprises to give increasing responsibility to business analysts. We also introduce the people who perform this essential role; we describe their education, professional backgrounds, daily responsibilities, and career paths.

Chapter 2: Conquering the Challenges of Business Analysis

Business analysts must overcome many hurdles in order to achieve maximum productivity and effectiveness. In this chapter, we look at each of the following challenges:

- ✓ Diverse software development methodologies
- ✓ Intricate business processes
- ✓ Inadequate tooling
- ✓ Cumbersome review cycles

Chapter 3: Building a Better Business Analysis Experience

Life doesn't need to be so difficult for the business analyst. Follow these three simple guidelines to dramatically improve the effectiveness of defining and managing requirements, and achieve project success:

- ✔ Get visual
- ✔ Get social
- ✔ Get going

Chapter 4: Introducing Blueprint

Blueprint is the preeminent vendor dedicated to delivering solutions that help define and manage software requirements. In this chapter, you find out more about Blueprint, its products, and how it has helped its customers improve their requirements definition and management.

Chapter 5: Ten Tips to Improve Business Analyst Productivity

Here you can find a helpful collection of general-purpose suggestions that you can use to make the life of a business analyst more pleasant — and productive.

Icons Used in This Book

Every For Dummies book features small illustrations, called *icons*, sprinkled throughout the margins. We use the following icons in this book.



The Tip icon guides you to right-on-target information to help you become a better business analyst.



The Remember icon highlights information that is especially important for you to know as you evaluate how to improve business analysis procedures at your organization. To determine the most important information in each chapter, look for paragraphs marked with this icon.



Seek out the On the Web icon if you want to find out even more about effective business analysis.

Where to Go from Here

You can read this book in any order that you want. What we mean is that we don't force you to start with Chapter 1 and read straight through Chapter 5. If you want to discover more about Blueprint, head to Chapter 4 first. Instead, if you want to explore the role of the business analyst, read Chapters 1, 2, and 3. Want some quick tips on how to improve the business analyst experience? Check out Chapter 5. How you use this book is entirely up to you.



You can also check out additional information online at www.blueprintsys.com/RDMforDummies.

Chapter 1

Meeting the Business Analyst

.....

In This Chapter

- ▶ Understanding today's business environment
 - ▶ Determining major stakeholders
 - ▶ Characterizing the business analyst
 - ▶ Predicting the evolution of the business analyst
-

Enterprises have never found software more indispensable, whether firing photon torpedoes, raising deflector shields, beaming up Scotty — or just conducting regular business operations.

The importance of specifying, developing (or acquiring), and implementing business applications correctly has increased in recent years. The heightened emphasis companies have placed on designing and rolling out software, combined with diverse factors such as competitive pressures, mergers and acquisitions, globally distributed teams, and technological advancements, all contribute to growing demands and responsibilities for a relatively new type of professional: the starship captain. No, we're kidding: this new professional is called the *business analyst*.

In this chapter, we tell you all about the trends in the strange new world of business and describe the vital role that the business analyst plays.

Boldly Exploring the Dynamic Business Environment

No matter what type of business you work in, your organization most likely relies on software to drive just about every aspect of daily operations. For decades, automation initiatives have focused on back-office software, such as applications dedicated to sales and marketing, accounting, human resources, and other internal functions. However, in recent years companies have also increased emphasis on customer-facing enterprise software solutions, such as websites, e-commerce portals, and self-service customer support, just to name a few.



Today's heterogeneous, multifaceted software inventories developed over many years. As a result, businesses must now maintain a hodgepodge collection of software assets from different origins and time periods. The company may have built some solutions in-house, external contractors may have built others, and the company may have purchased still others off the shelf from outside vendors. Lastly, some software may have arrived through mergers and acquisitions.

Given the ad-hoc way that systems come into being, they vary widely in terms of technology, design, quality, and maintainability. Regardless of the exact composition of your software portfolio, how well applications function — both as stand-alone products and when working together as part of a larger solution — heavily impacts your business.



Though the technology landscape has transformed over the years, the amount of change it has undergone pales in comparison with what has taken place in the overall business environment:

- ✓ **Economic realities:** Recent macroeconomic conditions place businesses under intense pressure to generate revenue, reduce cost, and at the same time deliver outstanding customer value.
- ✓ **Competitive pressures:** Businesses face a daily struggle against rivals new and old in much more dynamic and competitive marketplaces than ever before.

- ✔ **Outsourcing and distributed teams:** In reaction to today's shrinking margins and competitive pressures, many businesses turn to innovative cost-control and time-to-market strategies.
- ✔ **Regulatory requirements:** Governments and industry agencies alike are now much more energetic in their demands on business for governance and accountability.
- ✔ **Mergers and acquisitions:** The challenging business environment both forces and creates opportunities for business restructuring in the form of selling off business units or divisions, or of merging or acquiring entire companies.

As we describe in the introduction, the business analyst sits at the very heart of the enterprise's business software endeavors. Because software applications are now completely indispensable to business success, and because they operate in such dynamic environments, the role of business analyst has become essential and prominent. Before we look at who business analysts are and what they do, however, we first detail the other major players on a software project.

Engaging the Stakeholders

Whether you're building software from scratch, or implementing a packaged software product that you bought from a vendor, more participants — each with a highly diverse background — may be attached to your project than you first realize.

As the primary party accountable for the delivery of the new or enhanced business application, the IT organization enlists many different types of people on a software project, including the following:

- ✔ **Project manager:** Oversees all aspects of the initiative, right up to deployment into production.
- ✔ **Architect/Designer:** Establishes and maintains a solid technical foundation across the portfolio, ensuring the long-term viability of the new solution.
- ✔ **Developer:** Develops the application from the ground up, or integrates a third-party solution into the overall technology portfolio.

- ✔ **Tester:** Ensures the developed application fulfills its requirements and seeks out and identifies any flaws in the new system that may impact its ability to work as advertised.
- ✔ **IT operations:** Ensures that the solution is available and operational for everyone who needs access, after the solution has been built (or bought).

The business division also recruits a diverse talent roster:

- ✔ **Sponsoring executive:** Business owner of the overall initiative, and the person who usually signs the checks.
- ✔ **Client:** Person representing the needs of the business on a daily basis.
- ✔ **End-user:** Person, either internal to the organization or external, depending on the application, who will interact with the new solution.
- ✔ **Partner:** Internal employees and customers aren't the only stakeholders who participate on a new software project. In today's collaborative world, many firms rely on tight-knit relationships with external partners.
- ✔ **Government and regulatory agencies:** With so many enterprises coping with externally imposed constraints, these parties sometimes have seats at the table as well.



Any given project can involve 10 or more unique types of stakeholder, each of whom has a very different background, skillset, set of expectations, and obligations.

Although a modern software project involves so many players, each of the roles mentioned in the preceding list has been recognized as a distinct and separate job for many years by management, professional organizations, and vendors that design and sell specialized solutions to help each execute his or her role.

Yet one title is conspicuously missing from most roll calls: the business analyst. In fact, management has only recently recognized this role and designated a formalized title. A significant number of organizations still don't even use this label. What makes this state of affairs particularly ironic is that the business analyst plays a central role in coordinating the efforts of each of the different stakeholders on a project.



Business analysts did not have their own professional organization until fairly recently: the International Institute of Business Analysts (IIBA) didn't form until 2003.



To find out more about the IIBA, head to their website at <http://www.iiba.org>.

In response to the newfound attention organizations are paying to business analysts, vendors now market specialized solutions to them as well. As we explain in Chapter 2, these specialized solutions are essential to helping business analysts — and their projects — succeed.

Analyzing the (Business) Analyst

Although business analysts aren't all identical, they do tend to share a number of traits, characteristics, and common backgrounds.

Education

Business analysts are highly educated professionals and many hold university degrees. A notable percentage of business analysts go on to earn advanced degrees in fields such as computer science, engineering, and business administration.

Work background

Business analysts often transfer into their roles from other job categories. According to a recent survey, the five most common origins for business analysts are

- ✓ Software development
- ✓ IT operations
- ✓ Line of business
- ✓ Quality assurance
- ✓ User community

Place in the organization

In general, about half of the surveyed business analysts report to the IT division, with the remainder serving the line of business. Often those reporting to IT have a more technical skill-set (and sometimes have the title *business systems analyst*, or BSA) whereas those reporting to the business often have more domain knowledge and experience.

Given that business analysis is such a new and developing field, business analysts work on a diverse collection of teams within IT and business, such as:

- ✓ Centralized business analyst team
- ✓ Project/Program management office (PMO)
- ✓ Quality assurance and testing
- ✓ Software development

Responsibilities

Every day, the business analyst collaborates with a broad collection of stakeholders from the IT organization, the line of business, and outside participants such as customers, partners, and governmental entities. To succeed, the business analyst must be capable of understanding the distinct needs of each stakeholder and communicate with them in language that they understand.



At first glance, you may imagine that the business analyst focuses solely on defining and managing requirements, but this aspect forms only part of the story. In fact, many business analysts are also responsible for ancillary tasks related to these requirements, such as program and project management, testing, and profit and loss outcomes.

Although in some cases stakeholders are concentrated in one location, more often they are distributed across multiple locations. Dispersed stakeholders introduce additional complexities and challenges to the job of a business analyst, as we illustrate in Chapter 2.

Regardless of where the stakeholders operate, the business analyst must gather all relevant information, analyze it to produce a set of requirements, validate these requirements with the original stakeholders and sometimes other stakeholders as well, and then coordinate distribution of requirements information across a wide audience, such as:

- ✓ External customers
- ✓ IT operations
- ✓ Line of business
- ✓ Quality assurance and testing
- ✓ Software developers
- ✓ Technical architecture team

In many ways the role of a business analyst parallels that of an author writing a story or screenplay — either way, you could call it “The Voyages of the Enterprise.” Indeed, the business analyst must tell the story of the application that the business needs and do so in such a way that the business stakeholders can easily consume the story and confirm that it meets their needs, and that the technology stakeholders (that is, testers, architects, and developers) can read it and understand what they need to test, design, and develop. As with any good story, the business analyst must use appropriate language, add in clear illustrations, and provide good visual imagery.



Because many organizations have only recently identified business analysis as a separate and distinct job, business analysts often struggle with a lack of clearly defined and well-understood tasks, not to mention a scarcity of supporting technology.

Career trajectory

Organizations are now starting to recognize the importance of business analysis, and because of this development, new and exciting opportunities for advancement are opening up. As more companies adopt a *business analyst center of excellence* (CoE) or *requirements management office* (RMO) structure, they create well-defined career paths with manager- and director-level positions.

Software development environments

Enterprises employ an assortment of software development methodologies. Common labels for these approaches include the following:

- ✔ **Waterfall:** In this approach, a project flows through a series of rigidly defined, sequential phases such as requirements, design, development, testing, implementation, and deployment.
- ✔ **Incremental:** This approach starts out with a relatively small set of project features, and then builds upon this foundation by integrating new parts as they become available.
- ✔ **Spiral:** This is an incremental-based development approach with a heavy reliance on prototyping to reduce risk.
- ✔ **Unified process:** In this approach, a project relies heavily on use cases to drive incremental software delivery.
- ✔ **Agile:** This populist, developer-focused, highly iterative approach focuses on deliverable value versus interim work products.
- ✔ **Hybrid:** Enterprises can also create their own, site-specific blends of all of the above approaches. In particular, adaptations of agile for larger organizations are rapidly evolving; these are sometimes referred to as *enterprise agile*, among other monikers.

In the past several years, organizations have been more and more drawn to the agile approach to developing software. Agile usually incorporates three traits:

- ✔ Emphasis on speed, instead of time-consuming requirements definition
- ✔ Frequent software build/test/deploy cycles
- ✔ Constant feedback and adjustments

As we explain in Chapter 2, agile software development techniques can add to the pressure on business analysts.

Imagining the Future

The role of business analyst is continually changing. After all, the position is still relatively young compared to others in the organizational landscape.



Business analysts operate in an environment of great confusion and uncertainty, but this environment also yields tremendous opportunity to create a demonstrable, positive impact on the entire business.

For example, organizations increasingly expect business analysts to be able to understand, explain, and justify how the requirements that they help gather and organize deliver tangible value to the business. This trend alone will definitely have major impacts on the responsibilities of the business analyst. Business-level accountability will soon be an integral component of the business analyst job description, but satisfying these responsibilities will be particularly difficult given the obstacles business analysts already face each day.

In short the job of the business analyst is certainly a challenging one, but one that is rapidly increasing in importance, has lots of room to grow and innovate, and looks to provide tremendous opportunities. We are interested to see how the business analyst's role will change in the next generation of enterprise!

Chapter 2

Conquering the Challenges of Business Analysis

.....

In This Chapter

- ▶ Dealing with different development strategies
 - ▶ Keeping complicated business processes under control
 - ▶ Switching to specialized tools
 - ▶ Making review cycles more efficient
 - ▶ Seeing how inefficient business analysis affects real-world companies
-

As we describe in Chapter 1, enterprises, professional organizations, and vendors have long accepted and understood most of the assignments on any given technology project. They have, however, only recently recognized the separate and distinct job of the business analyst.

The challenges business analysts face are even more formidable because their daily responsibilities are in continual flux and undergoing ceaseless transformations. Why are their roles so fluid? Well, the answer lies in some sobering statistics about IT projects: end-users take advantage of only 45 percent of delivered features (according to the Standish *Chaos Report*), and rework consumes 40–50 percent of project budgets (Boehm and Basili). Issues that the project team could have addressed when defining requirements are key contributors to these unfortunate numbers. In fact, faulty requirements form the root cause for 75–85 percent of rework (Leffingwell). With the business analyst at the center of the requirements process, no wonder this role is attracting so much attention!



Mainly due to the pivotal nature of the business analyst role and its recent designation as a separate job responsibility, business analysts struggle to overcome numerous impediments. Some of the more pressing issues include:

- ✓ Diverse software development methodologies
- ✓ Intricate business processes
- ✓ Inadequate tooling
- ✓ Cumbersome review cycles

In the following section, we look at each of these obstacles in more detail, and then connect the dots to comprehend what happens to the entire organization when the business analyst can't surmount these roadblocks.

Coping with Diverse Software Development Methodologies

Businesses constantly search for the most modern and efficient techniques for designing, developing, testing, and deploying software. This focus isn't a surprise, especially given the strategic nature of information assets such as software and data: well-designed systems deliver demonstrable competitive advantages that directly translate to market share and profitability.

Over the years, many organizations have sought out, selected, and then implemented numerous different — and often contradictory — approaches to creating software, all in a frenetic attempt to seize the often-elusive strategic edge.



This has led to an abundance of incompatible techniques that the business analyst must find a way to reconcile. Business analysts often encounter fragmented teams using their own methods that can range from waterfall to incremental to highly iterative agile, not to mention internally created hybrids.

In recent years, the *agile* development approach has gained considerable ground, and now many organizations consider it a highly productive technique for delivering software. As you may expect from its name, agile means responding quickly to changes. Agile arose from developer dissatisfaction with rigorous, prescriptive, heavy processes that, in the view of these developers, focused on the wrong elements.

Agile is not just one methodology but a family of methodologies that sprung from a common set of principles called the *Manifesto for Agile Software Development*, written by a group of 17 developers in 2001 (see agilemanifesto.org). The manifesto states the following:

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

Coordinating teams

Agile teams are typically comprised of four to nine people who play specific roles, although who plays each role can change. These teams control planning and prioritization for their area of work, and they connect on a daily basis in an interactive session called a *scrum* to coordinate and synchronize their efforts.



One of the most popular agile methodologies is called *scrum* and its traits are consistent with the *Manifesto for Agile Software Development*.

Sprinting to milestones

Development proceeds in well-defined time periods called *sprints*, which typically last two or three weeks and the result of each sprint is a functioning, demonstrable portion of the product.

Clearing the backlog

The essence of an agile approach like scrum is its ability to respond effectively to changes in priorities during the project. The project backlog is a prioritized list of the work to be done. The end of every sprint presents the opportunity to change direction. In other words, you can start developing the next sprint to a revised set of priorities as defined in the backlog. You can see an overhead view of this process in Figure 2-1.

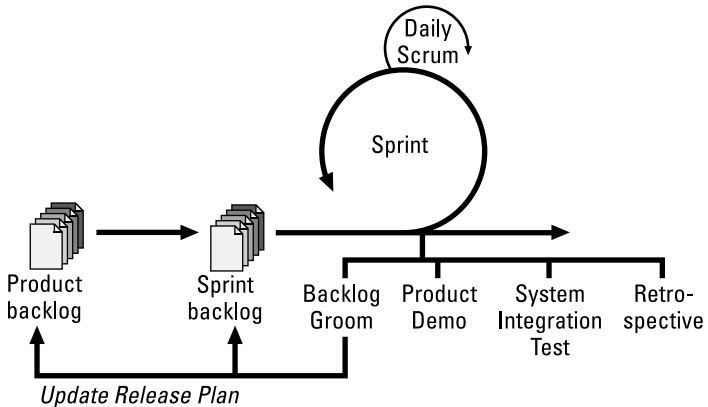


Figure 2-1: Agile at work.

Agile is particularly daunting for the business analyst because of the marked reduction in the amount of time spent on — and attention paid to — the job of authoring, understanding, reviewing, and approving requirements before the scrum teams begin to develop the system. In fact, many cite the

alleged efficiencies this reduction causes as *the* best reason to switch to an agile approach. Yet advocates of agile often fail to recognize that paying less attention to these important areas can cause havoc and chaos, waste time, and squander money.



Good requirements and agile are absolutely compatible — in principle. Requirements are authored in agile prior to building iterations of the system. These requirements are detailed for each iteration individually to a level consistent with other methods just before implementation.



Check out the Agile Alliance, an agile advocacy group, online at www.agilealliance.org.

Managing Convoluted Business Processes

The business analyst can properly and thoroughly author and document requirements for any new software solution only by fully comprehending the underlying — and often extremely complex — business processes that will be supported by the new system. Because a project is full of unknowns, the business analyst must stay on top of these processes throughout the project. Speaking of rapids, in waterfall methodologies where the project team creates a more exhaustive and detailed set of requirements up front, the lessons learned along the way are incorporated later in the form of enhancement requests. Agile, however, follows more of a learn-as-you-go approach. The business analyst position is daunting enough when you examine it at face value, but a number of factors combine to make it even trickier:

- ✓ **Business processes are always evolving:** Given how fast operational procedures can change, the initially defined requirements may quickly go stale.
- ✓ **IT and line-of-business stakeholders are dispersed:** Conducting effective research is hard enough when all stakeholders — each with his or her own set of needs and agendas — can gather around a single conference



room table. The task is much more imposing when the stakeholders are distributed around the world, where they often work for different organizations. You can see a visual depiction of distributed teams in Figure 2-2.

- ✓ **The business itself may undergo rapid change:** Very little is permanent today; regulatory changes, mergers, acquisitions, and sudden demands for more rigorous cost control can affect every enterprise.

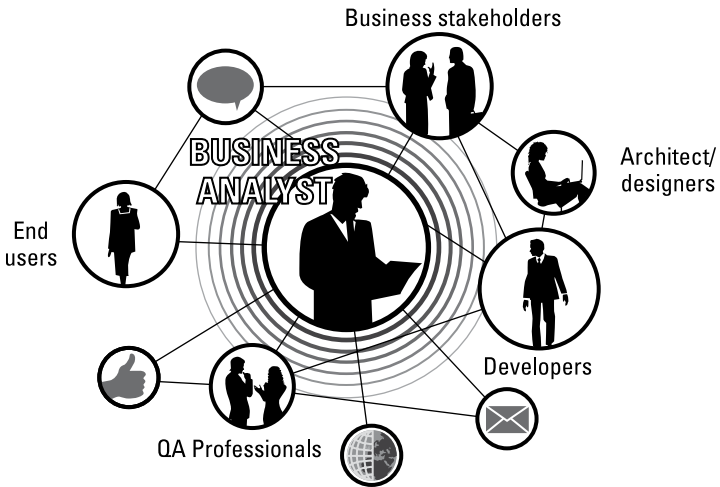


Figure 2-2: Business analysts are at the heart of distributed team collaboration.

Not only are business analysts confronted with agile software development techniques that devalue their work, and highly involved business processes that change continuously, but subpar supporting technology also hampers many business analysts — which is what we describe next.

Overcoming Inadequate Tooling

According to the *voke Market Snapshot Report: The Role of the Business Analyst*, 80 percent of business analysts still carry out their highly challenging jobs using generic, non-specialized personal productivity tools:

Typing when you don't need to

The act of writing a document consumes an unnecessary amount of time, especially for a detail-oriented procedure such as designing a software solution. Ironically, as the review cycles plod on, business analysts feel compelled to write additional text to provide more clarity, which makes the requirements document even bigger, takes more time, and further expands the review cycle.

Doing clerical work that's not your job

Shepherding and organizing documents and review comments is primarily a non-creative and time-consuming set of rote chores. Focusing on these tasks doesn't leave a lot of time for the primary, strategic value of the business analyst: critical and analytical thinking. Getting better at your job is nearly impossible when you're busy putting out fires and working with inefficient tools.

On top of all these headaches, the true implications of subpar tooling really become evident during the review cycle, particularly when team members use general-purpose tools that are not aimed at supporting the distinct needs of business analysts.

Conquering Cumbersome Review Cycles

After the first pass at creating a requirements document is done, other project participants often give review cycles a low priority. They don't push the review process aside because they are purposely ignoring their job responsibilities; they do so because the act of reviewing a massive document is tedious at best. Indeed, reviewing a massive document sets their faces to *stunned*. Furthermore, they may not even notice the document: if you email vital requirements documents they compete with all sorts of other email clutter for reviewers' attention.



The review-and-edit cycle consumes an incredible amount of time, which can grow exponentially as the size of the document increases. In fact, the bigger the document, the more likely that the reviewers simply sign off without reading it. These blind approvals can create serious problems later on.

When reviewers actually do examine documents, the track changes capability of word processors doesn't encourage collaboration. Instead, reviewers examine requirements documents in isolation and clutter them even more with multi-colored markups. See Figure 2-4 for an example.

CHAPTER 2. GENERAL REQUIREMENTS

JobDescription This element describes the job and its requirements. It contains **JobIdentification**, **Application**, **Resources**, and **DataStaging** elements.

JobIdentification This element contains all elements that identify the job: **JobName**, **Description**, **JobAnnotation**, and **JobProject**. If this element is not present then its value, including all of its sub-elements, is undefined and will not be initially defined when the system is instantiated.

Application (This element describes the application and its requirements). It contains **ApplicationName**, **ApplicationVersion** and **Description** elements. It serves as a high-unique descriptor for the low-level generic contains-episode that is intended to hold more specific application definitions. One such definition is listed in the POSIX compliant normative extension given in [JSDL], §8.1.1. Used without any extension, it unambiguously describes an application by its name and version number. If this is not present then this job definition does not define a module or component application to execute. The JSDL 7.9.2 document could be defining a data staging job, or a null job. See also [JSDL], §6.3.2 and §8.1.2.

Resources This element contains-describes the resource requirements of the job being operated on. If this element is not present-present or contains a null value then the consuming system may choose any set of resources to execute the job.

Any combination of the listed resources sub-elements may be present in the resources element of a JSDL-KLONE document. In particular, any combination of "Individual", and "Total" elements of the same or different types may be present in a resource element. But, note that all elements present in a JSDL document must be satisfied for the entire document to be satisfied. (See also [JSDL], §4).

DataStaging Data staging defines the significant files that should-must be moved to the execution host (stage in) and the files that should be moved from the execution host (stage out). Files are staged in before the job starts executing. Files are staged out after the job terminates.

If a directory is specified in the **FileName** element or **Source** element then a recursive copy will be performed until the limits specified in HUB9 specification are reached. If the execution environment does not support recursive copying an error should be reported. The specification of this error, including how or when it is raised, is beyond the scope of the JSDL specification.

It is not possible to stage out the same file more than once by specifying the same **FileName**, (on the same **FileSystem**) in multiple stage out **DataStaging** elements.

Formatted: Font Bold, Underline

Comment [M1]: This is not a valid element in the system. We had this discussion last week during the client meeting and they agreed. In the K090 the elements were all described there and nobody could argue on whether or not they should be included. We should bring this up in the next review session but if P030 doesn't hear, then it shouldn't be captured as one of the essential elements.

Comment [M2]: There is uncertainty here based on some discussions with the system architecture team. Need to re-communicate.

Formatted: Highlight

Formatted: Right: 0.66 cm, Space Before: 0.9 pt, Line spacing: single

Comment [M3]: Need to account for null value states.

Formatted: Underline

Comment [M4]: Not all the information in here is correct. I think we spoke about this three days ago and nothing has changed since then to cause a reconsideration.

Comment [M5]: Need to have development confirm this is possible. Based on an email thread, I have some doubts to this being technically feasible.

Comment [M6]: Where is this term defined?

Formatted: Font Bold, Italic, Underline

Comment [M7]: Why is not possible? Please elaborate.

Figure 2-4: The inefficiencies of using track changes.

When it comes to dependencies and relationships among the requirements (traceability), business analysts must track them using spreadsheets, which are notoriously inefficient for performing these types of tasks. Compounding the problem further, spreadsheets don't include an automatic history of changes or evolution of the requirements.



Design documents, emails, and dependency tracking spreadsheets aren't linked together, which means that team members can easily lose the thread and forget how the final design came together.

Exploring the Economic Risks of Inefficient Business Analysis

Aside from making the business analyst's job much more difficult, significant financial and operational risks directly result from the inefficient ways that organizations create and manage requirements definitions. Risks include not only wasting money on the project itself, but also the economic impact of not solving the problem that the project is supposed to address. Often this latter risk is orders of magnitude worse than the expenditures on the project.



For example, consider these real-world crises that failed software projects caused directly:

- ✔ **Ford Motor Company:** Ford abandoned a \$400-million-dollar purchasing system. To find out more, visit <http://trunc.it/n0qf7>.
- ✔ **New York City:** The CityTime payroll system ballooned significantly over budget. It began as \$63 million dollar project, but ended up costing \$760 million. You can read all the details at <http://trunc.it/mji48>.
- ✔ **Internal Revenue Service:** The IRS abandoned a highly touted electronic fraud detection system after a total expenditure of \$185 million. This cost makes up only part of the damage, because the lack of a solution to fraud issues led to more than \$800 million worth of fraudulent refunds in the first year alone. Visit simplearchitectures.blogspot.ca/2009/09/cost-of-it-failure.html and search for **Fraud** on the page.

- ✓ **UK Electronic Health Records:** The Department of Health abandoned the entire program after a \$12-billion expenditure. You can find a brief overview here: www.ihealthbeat.org/articles/2011/8/5/uk-health-system-expected-to-abandon-ehealth-network.aspx.
- ✓ **U.S. Air Force Expeditionary Combat Support System:** The Air Force cancelled the project after cost projections for the project rose from \$1-billion to \$2.1 billion and delivery projections climbed to eight years beyond the original date. Investigators cited requirements as one of the root causes. Read the full story here: <http://www.bloomberg.com/news/2013-01-24/senators-to-probe-air-force-s-1-billion-failed-software.html>.

Chapter 3

Building a Better Business Analysis Experience

In This Chapter

- ▶ Borrowing tactics from other disciplines
 - ▶ Making information visual
 - ▶ Collaborating with your team
 - ▶ Overcoming misconceptions that block progress
 - ▶ Predicting changes in business analysis
-

In Chapters 1 and 2 of this book, we describe all the negative circumstances that business analysts must transcend. But neither hodgepodge software assets nor convoluted business practices can stay these professionals from the swift completion of their appointed duties: proven techniques can make the business analysis process much smoother, more efficient, and more pleasant for every stakeholder — which benefits the entire organization, not just the business analyst.

The first step on the road to this happy destination is to realize that business analysts aren't the only professionals who must define complex systems. In fact, you can discover a lot by examining how other trades do it.

Taking Strategies from Other Industries

The failures and abandoned projects that are so commonplace in IT just aren't tolerated in other endeavors, such as:

- ✓ Architecture
- ✓ Construction
- ✓ Manufacturing
- ✓ Transportation

After all, how often do you hear of an abandoned, half-constructed skyscraper?



How do designers and implementers in each of these trades communicate? Here's a hint: they don't generally write 50-pound specifications to get their ideas across. Instead, they use visual representations whenever they can. These blueprints and models communicate highly complex concepts in as streamlined a manner as possible.

They must be making some good choices, because buildings take shape and remain standing, and airplanes roll out onto the tarmac and fly.

IT organizations may not get to smash champagne bottles on the front of a new software release, but who says they can't adopt these time-tested approaches to generating and disseminating blueprints?

Three major (yet surprisingly simple to follow) strategies make the job of creating software requirements easier:

- ✓ Getting visual
- ✓ Getting social
- ✓ Getting going

Getting Visual

As we point out in Chapter 2, enormous text-based specifications are not the most effective way of communicating the

complex concepts that underlie modern software solutions. In fact, these massive documents may be the worst way to present this type of information, especially because line-of-business and IT organizations don't commonly use the same vocabularies to describe the same concepts. You can see a sample of the types of terms used by business and IT staff in Figure 3-1. (But keep in mind that the words in the left column don't match the words in the right column. Don't start casually dropping the word *widget* into discussions with IT staff believing that it's a synonym for *earnings per share*.)

Business Terms	IT Terms
Balance Sheet	Compiler
Bankruptcy	Deprecated
Debt-to-Equity Ratio	Runtime
Value Line Index	SDK
Yield	Query
Trading Range	Memory Leak
Put Option	Multireading
Revenue Bond	Iteration
Operating Margin	Encoding
Market Capitalization	WYSIWYG
Earnings Per Share	Widget
Rate of Return	Syntax
Compounding	Unmount
Transaction Cost	Spool
Market Order	Root
Buy-and-Hold	Paste
Bid Price	Java
Total Liabilities	Kernel
Dividend	DBMS
Holding Period Return	Virtualization
Prospectus	Template
Present Value	Token
Liquidity	Pseudo Code
Stop-Loss Order	Interpreter

Figure 3-1: Example vocabulary differences between business and IT



Instead of writing the next great novel, why not use the proven power of pictures, diagrams, flowcharts, user interface mockups, and other well-regarded graphical techniques to help arrive at a consensus about how you want the new system to work?

Switching from heavy reliance on text to a more balanced approach using visual images delivers a number of attractive benefits:

- ✔ **Clerical work doesn't take over:** Business analysts no longer perform the largely clerical role of maintaining an unwieldy text document. Instead, they can apply their critical thinking and analytical skills to help deliver better solutions.
- ✔ **Language barriers disappear:** Visual representations aren't burdened by perplexing cross-department or spoken language differences and are thus far easier to understand. A common visual language helps bring the proposed system to life.
- ✔ **Review cycles go faster:** Clear, more visual representations help to greatly reduce the amount of time spent during the review cycle, and make reviewers more likely to actually want to engage.
- ✔ **Bottlenecks expand:** When the business analyst no longer has to perform meticulous document maintenance, he or she is no longer the main impediment to progress in the review cycle.

Taken together, all of these improvements bring clarity and efficiency to the software process — including agile. Examples of these visualizations are shown in Figure 3-2.

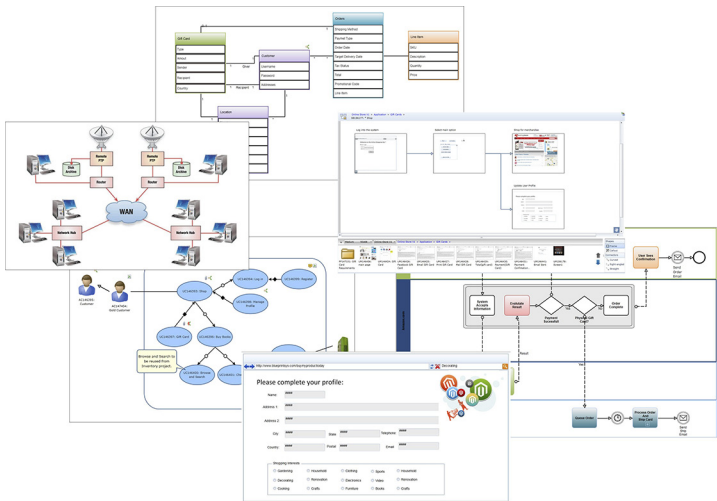


Figure 3-2: Taking a visual approach

Getting Social

The most effective efforts arise when diverse and talented teams collaborate, yet many software development projects do the opposite.

We can understand why team members find it hard to collaborate; look at all the barriers they must overcome:

- ✔ Distributed teams across separate time zones, languages, and cultures
- ✔ Insufficient tooling, reflected by enormous specifications that are assembled in documents and emailed everywhere
- ✔ Many different job functions
- ✔ Never-ending review cycles
- ✔ User resistance to endless review sessions



Despite what word processor–software vendors tell you, employing the track changes feature (shown in Figure 3-3) on thick, heavy documents doesn't facilitate well-integrated, productive, and harmonious team collaboration, especially when your work incorporates complex and interrelated concepts.

Instead of continuing to suffer alone being tied to track changes, why not incorporate some proven social-based techniques to get people working together?

Historically, requirements definition efforts have generally gone onward with relatively little interaction primarily because the supporting tools weren't mature enough, resulting in practitioners spending all their time with their heads down working in silos.

CHAPTER 2. GENERAL REQUIREMENTS

JobDescription This element describes the job and its requirements. It contains JobIdentification, Application, Resources, and DataStaging elements.

JobIdentification This element contains all elements that identify the job: JobName, Description, JobAnnotation, and JobProject. If this element is not present then its value, including all of its sub-elements, is undefined will not be initially defined when the system is instantiated.

Application This element describes the application and its requirements. It contains ApplicationName, ApplicationVersion and Description elements. It serves as a high-unique descriptor for the low-level generic container-episode that is intended to hold more specific application definitions. One such definition is that of the POSIX compliant normative extension given in [JSDL], § 1.1. Used without any extension, it uniformly describes an application by its name and version number. If this is not present then this job definition does not define a module or component application component to execute. The JSDL 792 document could be defining a data staging job, or a null job. See also [JSDL], § 6.3.2 and § 8.1.2.

Resources This element contains describes the resource requirements of the job being operated on. If this element is not present present or contains a null value then the consuming system may choose any set of resources to execute the job.

Any combination of the listed resources sub-elements may be present in the resources element of a JSDL KLOPD document. In particular, any combination of "Individual", and "Total" elements of the same or different types may be present in a resource element. But note that all elements present in a JSDL document must be satisfied for the entire document to be satisfied. (See, also [JSDL], §4)

DataStaging Data staging defines the significant files that should must be moved to the execution host (stage in) and the files that should be moved from the execution host (stage out). Files are staged in before the job starts executing. Files are staged out The modules being considered will terminate after the job terminates.

If a directory is specified in the FileName element or Source element then a recursive copy will be performed until the limits specified in HJ89 specification are reached. If the execution environment does not support recursive copying an error should be reported. The specification of this error, including how or when it is raised, is beyond the scope of the JSDL specification.

It is not possible to stage out the same file more than once by specifying the same FileName (on the same FileSystem) in multiple stage out DataStaging elements.

Formatted: Font Bold, Underline

Comment [M1]: This is not a valid element in the system. We had this discussion last week during the client meeting and they agreed. In the K990 the elements were all described there and nobody could agree on whether or not they should be included. We should bring this up in the next review session but if PU90 doesn't fix it, then it shouldn't be captured as one of the essential elements.

Comment [M2]: There is uncertainty here based on some discussions with the systems architecture team. Need to re-evaluate.

Formatted: Highlight

Formatted: Right: 0.66 cm, Space Before: 0.9 pt, Line Spacing: single

Comment [M3]: Need to account for null value states.

Formatted: Underline

Comment [M4]: Not all the information in here is correct. I think we spoke about this three days ago and nothing has changed since then to cause a reconsideration.

Comment [M5]: Need to have development confirm this is possible. Based on an email thread I have some doubts to this being technically feasible.

Comment [M6]: Where is this term defined?

Formatted: Font Bold, Italic, Underline

Comment [M7]: Why is it not possible? Please elaborate.

Figure 3-3: The inefficiencies of using track changes

But now solutions are available on the market that blend the visual approaches we describe earlier in this chapter with sophisticated yet easy-to-use collaboration features such as:

- Inline discussions that traverse departments and time zones
- Easy online self-service review and approve

See the inline discussions of a streamlined review cycle in Figure 3-4.

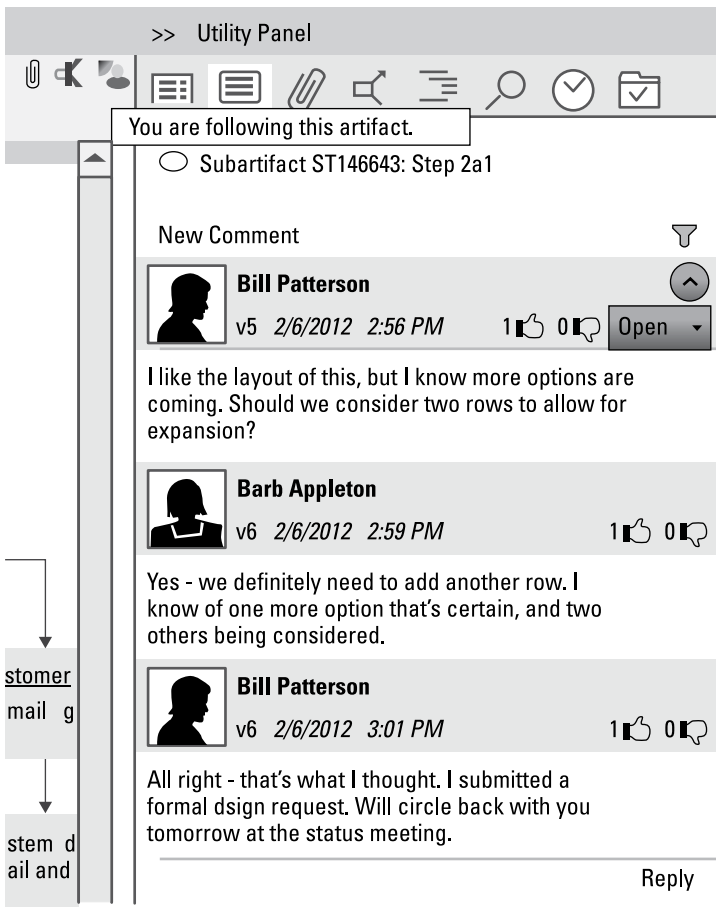


Figure 3-4: Inline discussions in a streamlined review cycle.

A streamlined review cycle lets stakeholders decide how, when, and from where they want to participate, instead of being caught and dragged into meetings.

Getting Going

Two perceptions among IT organizations have formed barriers to business analysis progress.

Reengineering processes

To begin with, many organization members believe they must first reengineer all of their processes before bringing in supporting technology. This belief just isn't accurate — it's a myth that must be busted.

With such little investment made in business analysis tooling in the past, the new crop of products represents a huge leap forward in capability. These capabilities make possible processes that organizations otherwise may not even consider, so crafting a process without regard for the tools can result in missing these opportunities. Instead of waiting for the perfect processes to arrive, why not employ technology up front to help improve these critical processes in a more evolutionary fashion? Having the right tool allows organizations — for the first time — to make specific process improvements that they can then use to drive further refinements and new process definitions. Thus, deploying the right tooling serves as a catalyst for many future optimizations.

Rolling out software

The other misconception IT organizations have is that if they implement powerful new technology they must make wholesale changes to the way they conduct daily operations. Many new solutions aimed at the business analyst, however, are perfectly capable of rolling adoption: certain features are enabled up front, and others are incorporated as time and business needs require.

After all, most users of Microsoft Excel employ only about 5 percent of its full capability, yet they still derive tremendous value from it.

You can find out more details about the Blueprint software solution in Chapter 4, but see Figure 3-5 for the possibilities of a phased rollout of a requirements definition and management solution.

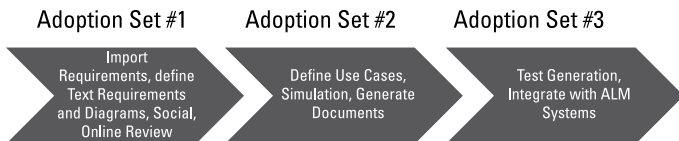


Figure 3-5: Sample steps in a phased rollout from Blueprint.

Forecasting the Future

By now you can see that the business analyst performs a highly dynamic, subjective job — to say the least! Where is the business analyst role heading? We see at least four trends that will continue to impact the business analyst:

- ✔ The role will become even more formalized and influential as companies realize its importance to the organization.
- ✔ More will be expected of business analysts as they will increasingly take responsibility not only for feature-laden new solutions, but also for the business value realized from them.
- ✔ Business analysis *centers of excellence* (CoEs) and *requirements management offices* (RMOs) will emerge and grow in prominence as they increasingly prove their value.
- ✔ Business analysts will become pivotal for large enterprises currently struggling to adopt agile practices as they recognize the key role requirements play in larger, distributed initiatives.

Chapter 4

Introducing Blueprint

In This Chapter

- ▶ Checking out the company's background
- ▶ Exploring the solutions Blueprint offers
- ▶ Observing Blueprint in action

As we depict in Chapter 1, management, professional organizations, and vendors are finally giving business analysts the recognition and respect they deserve. Blueprint is one of the most prominent solution providers to service these previously ignored professionals.

In this chapter, we introduce you to Blueprint's product, also named "Blueprint," which was designed specifically for business analysts. We begin by describing the background and founding philosophy of Blueprint, followed by an overview of its product. We pay particular attention to how you can put Blueprint to work to overcome the barriers to success that have been frustrating business analysts for years. We then showcase several examples of Blueprint delivering tangible benefits to its clients.

Highlighting the History of Blueprint

Blueprint was founded in 2004 and was a pioneer in the emerging discipline of requirements definition. Since that time, Blueprint has focused entirely on providing solutions to eliminate the problems associated with defining and managing software requirements.



According to industry data, creating or enhancing business applications is fraught with budget and schedule overruns. This process also heavily influences application quality. Many of the issues organizations experience stem from poor or inadequate levels of interaction between business stakeholders and the IT professionals developing the software.

Blueprint can transform this business-IT relationship into a visual and engaging collaboration. Errors and omissions surface earlier, team members deliver a higher-quality set of application requirements sooner, and requirements-related rework — which studies show consumes 40–50 percent of project budgets on average — shrinks. Blueprint’s unified approach improves the probability of delivering applications on time and on budget. Predictable project schedules combined with faster time to market is critical to the competitive success of Blueprint’s global customer base, and these customers depend on Blueprint to achieve their goals.



Find out more about Blueprint here: www.blueprintsys.com.

Sorting through Blueprint Solutions

In this section, we tell you more about how Blueprint solves the difficult requirements definition and management problem. To help bring the discussion to life, we examine the impact of Blueprint on several of the most essential job responsibilities of the business analyst.

Authoring requirements

As we describe in Chapter 3, business analysts must continually strive to replace words with visual representations when creating requirements. Doing so helps to remove ambiguity and make the resultant requirements more engaging and easier to understand.

Blueprint provides a broad range of rich, visual requirements editors, as shown in Figure 4-1. You can choose the format that's perfectly suited to each type of requirement, which promotes consistency and ensures that you don't miss any important elements.

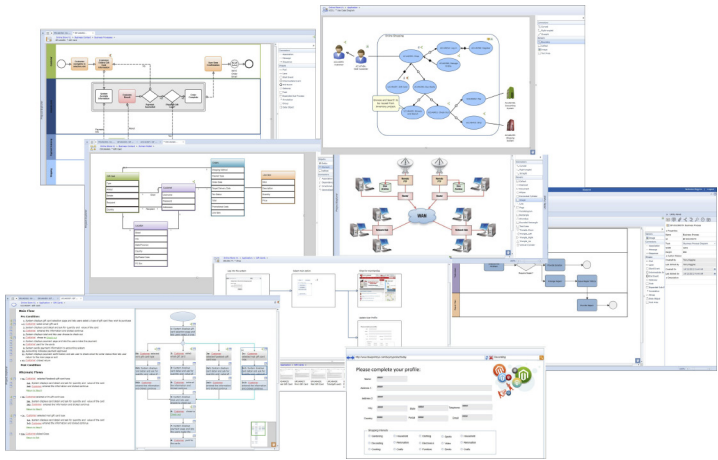


Figure 4-1: Blueprint provides a wide range of visual requirement editors.

Validating requirements

Creating a well-defined set of requirements is just the start: making sure that they are correct is equally important. Blueprint engages stakeholders and helps them confidently drive a set of requirements to final approval. Blueprint's online review and approve interface makes reviewing requirements fast and easy for stakeholders, while review dashboards let the business analyst control, coordinate, and conclude large-scale requirements reviews among the diverse set of stakeholders that are so common on today's software projects.

Blueprint's rich visual requirements simulations keep stakeholders engaged and let them understand the true meaning of the requirements — a fundamental prerequisite for obtaining better feedback. Blueprint also automatically generates

requirements documents that support existing company processes and provide the hard-copy artifacts that are often needed for official sign-off. To get a better idea of the Blueprint software's approach to interactive online review, approvals, and requirements simulations, see Figure 4-2.

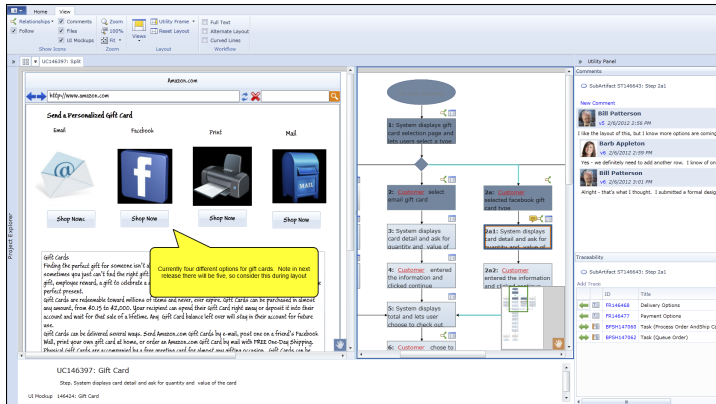


Figure 4-2: Blueprint's visual and interactive requirements simulation.

Managing requirements

Even after you've authored and validated your requirements, you must still maintain them, track their histories, and keep tabs on how they evolve over time. Blueprint was designed with the entire requirements lifecycle in mind.

To begin, Blueprint automatically versions every requirement, which makes it possible to examine each requirement's entire history beginning with the baseline. Next, Blueprint quickly creates traceability that's visible while working and is easy to analyze through a suite of powerful tools. Traceability helps you assess the impact of change, confirm coverage, and manage scope. Figure 4-3 shows the fine-grained traceability features of Blueprint.

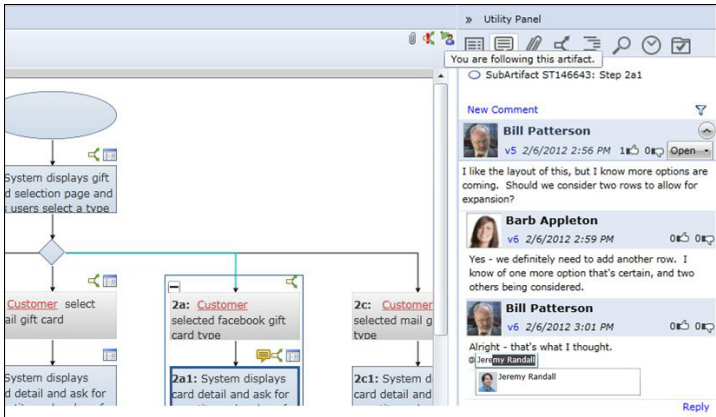


Figure 4-4: Threaded discussions with @mention.

Integrating with other technologies

In Chapter 2 we demonstrate that the business analyst sits at the center of the entire software project lifecycle. Therefore, for maximum effectiveness, any business analysis tools must tie in with technologies used by other stakeholders.

Blueprint incorporates practical integrations with the most popular tools project team members use on application development projects. It automatically generates a comprehensive set of tests from the requirements, and then populates *application lifecycle management (ALM)* solutions with those tests and the underlying requirements — including complete traceability. Figure 4-5 shows Blueprint generating test cases.

Blueprint also provides integration with Microsoft Excel that lets users directly leverage the spreadsheet software's many features, or use it as a conduit to other third-party toolsets. These integrations help ensure alignment of all groups who collaborate on application projects.

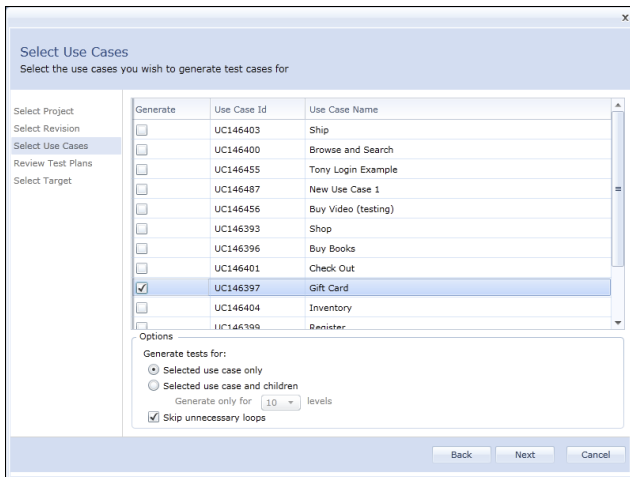


Figure 4-5: Generating test cases with Blueprint.

Watching Blueprint at Work

In this section, we take you through a few examples of what Blueprint has been up to. We detail some real customer examples, and show you how Blueprint has transformed the requirements definition and management process, along with demonstrating the meaningful impact this transformation has had on overall business performance. Getting blue isn't always so bad!

Banking industry solutions

A large global banking institution embarked on a multiyear program to upgrade their core banking systems. All of the over 1,000 people (distributed across 26 countries) in the program who are authoring or collaborating on requirements are using Blueprint. To enable such a large group of users, and sustain them as staff and outsourcers change over time, the institution is leveraging the online e-learning component of Blueprint, allowing any number of users to learn at their own pace and their supervisors to track their progress.

Legal industry solutions

A prominent solution provider to the legal industry had to transition its flagship software product to a more modern platform and enhance it with significant new capabilities. The organization turned to Blueprint as the requirements application to define and manage this business-critical effort. Due to the success of this initiative, the company has since standardized on Blueprint as the requirements application used on all IT software projects, enterprise-wide.

Health industry solutions

One of the largest health insurance organizations in the United States recognized it was experiencing too much rework, and reengineering, as well as poor requirements, and organization members knew they were not taking advantage of reuse opportunities in the area of requirements. In this environment, the organization had a mandate to update their systems to be ICD-10 compliant, a major undertaking across their systems, before October 2013. After extensive research they selected and deployed Blueprint, and formalized their requirements processes around it as their standard requirements platform. Today their development projects are far more predictable, reuse and efficiency has increased, and they are well on their way to attaining ICD-10 compliance.

Legacy modernization solutions

A very large pension-provider organization needed to modernize a business-critical application hosted on a mainframe system. The application had to adhere to strict pension legislation or risk substantial penalties. The organization's existing requirements processes didn't provide good visibility into the as-is state of the system, putting all modernization work at huge risk — and processes were so onerous that people regularly subverted them to get their jobs done. After deploying Blueprint, communications and collaboration around requirements improved to the extent that stakeholders now see value in business analysis where before they had seen only overhead. The as-is state of the system is now modeled for all to analyze and understand. As a result, the application was delivered on time and now operates in full compliance.

Chapter 5

Ten Tips to Improve Business Analyst Productivity

.....

In This Chapter

- ▶ Putting the power of pictures to work for you
 - ▶ Including your stakeholders every step of the way
 - ▶ Making life easy for your developers, your testers, and yourself
-

Throughout this book, we tell you about the ever-changing role of the business analyst, the challenges business analysts face, and how they can take steps to make projects go more efficiently. This chapter puts all of this information to work by presenting you with a series of field-tested guidelines and best practices.

We begin by offering some suggestions about requirements authoring, followed by several recommendations about how to more effectively work with all of your stakeholders. Finally, we provide a collection of our thoughts regarding smooth collaboration with the people charged with bringing your vision to life: architects, developers and testers.

Leveraging Visual Models

In Chapter 3 we describe how diverse professions ranging from architecture to mechanical engineering all use visualizations wherever possible. Visual models are absolutely critical

if you're going to do good analysis, and especially when defining complex solutions such as software. Get into the habit of turning to visual representations whenever you can.



Want more details? Check out Joy Beatty's and Anthony Chen's book *Visual Models for Software Requirements*, here: <http://trunc.it/n5x28>.

Expressing Your Requirements

Every business analyst has the potential to do a great job working with their stakeholders to create very expressive visual models, but that's only part of the story. These graphically rich artifacts are also superb for quickly and cleanly getting information across to others such as developers, testers, support and operations staff, and more, as you fully define the requirements for your new solution. See an example in Figure 5-1.

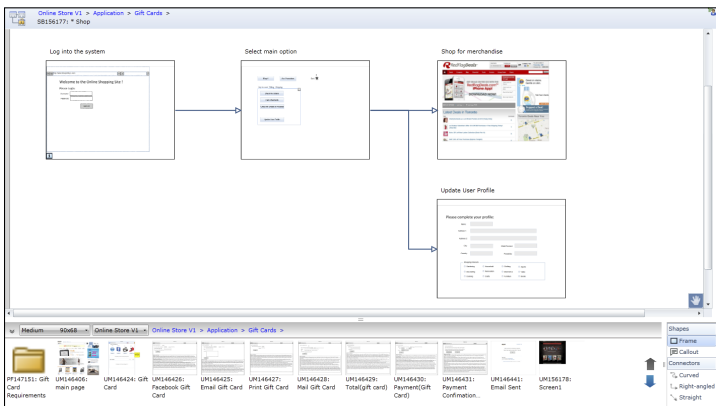


Figure 5-1: Expressing requirements visually.

You can use these models by themselves, or reference them in requirements.

Setting the Stage for Effective Collaboration

Healthy partnerships don't happen by accident. Instead, they result from establishing the communication channels that you want people to use, and then making sure these conduits are accessible and efficient. If you fail to take these steps, people get frustrated and simply bypass your painstakingly created infrastructure. The end result is a flood of inefficient point-to-point interactions. Blueprint provides an easy way to collaborate, as shown in Figure 5-2.



Figure 5-2: Multi-party collaboration with Blueprint software.



Whatever mechanisms you set up to foster collaboration, they need to be available 24/7 and worldwide to better support your highly distributed teams.

Analyzing Stakeholders

All stakeholders are not created equal. In fact, some are more influential than others. Therefore, the business analyst must

first identify all of the relevant stakeholders for the project, and then determine which of these parties outrank the others. Failing to do so can derail a project before it even gets started.



Initially, start with the stakeholders funding the project. These stakeholders give a good perspective of the *intent* of the application. Another stakeholder group comes from governance, compliance, risk, or other regulatory bodies; these people will provide the *governance* required for the application as it performs the intended role. Another group of stakeholders focuses on the user roles of the application and provides the *operational* perspective. Every requirement that ultimately reaches the final set must be within the project's intent, meet the required governance demands, and provide value to the user operationally.

For example, imagine gathering needs for a flight reservation application. You specify the requirements and initiate development but neglect to involve the employees who are in the midst of modifying the company's operating processes to adhere to the latest changes in regulations. Then you find out late in the development that some of these changes directly impact the new flight reservation system. This scenario can be incredibly expensive and delay rollout dramatically as team members make unplanned changes (perform rework). If you involve those stakeholders early in the process, however, you can consider the regulatory process changes in the initial requirements definition.

Getting the Right Tool for the Job

As with any trade, if you don't have the right tool your work suffers.



As we describe in Chapter 2, many business analysts attempt to leverage and even extend or customize, general-purpose tools such as Microsoft Office to satisfy their unique

requirements. Generally this tactic is highly inefficient, and it can consume an enormous amount of time that can be better spent focusing on your core responsibilities. Want to see an effective requirements application in action? Check out Blueprint in Figure 5-3.

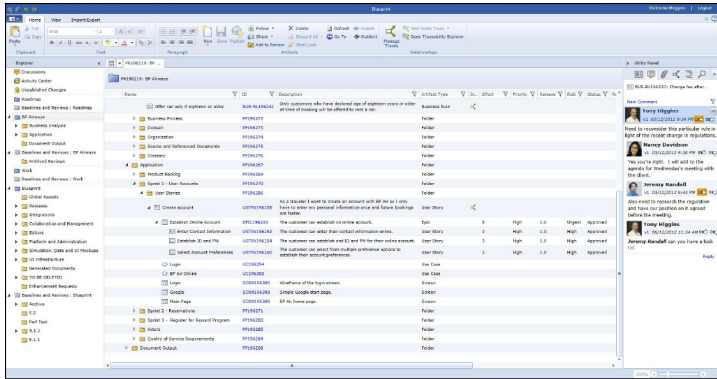


Figure 5-3: Blueprint.

Avoiding Being a Slave to the Document

Like any other massive piece of work, a requirements document can take on a life of its own. Once it breaks its bonds and staggers to its feet, it quickly becomes the focus of attention, and saps incredible amounts of energy just to keep it organized and current.



Focus on the document's content — the requirements — not the container. And instead of relying too heavily on text, strive for an approach that uses a blend of images, diagrams, and words (see how Blueprint does it in Figure 5-4). The ultimate goal is to prove that you have a clear definition of your business solution. Any approach that achieves that objective the most clearly is the right approach.

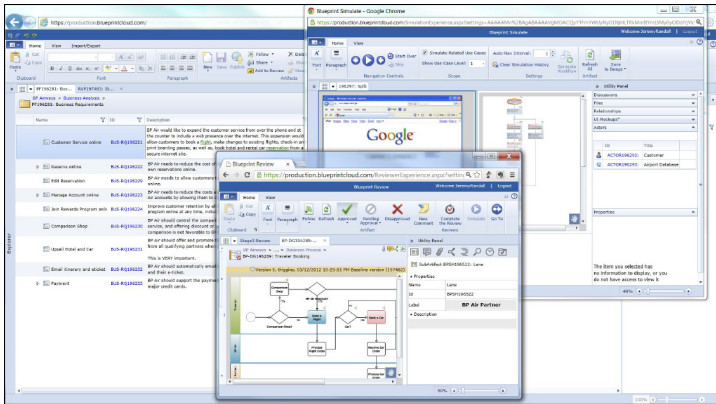


Figure 5-4: The right approach: a blend of images, diagrams, and words.

Creating a Traceability Strategy

The only constant is change with respect to business analysis: solutions inevitably evolve over time. Setting up a well-designed traceability strategy yields valuable benefits, such as:

- Accurately managing change requests
- Guiding and controlling your project's scope
- Measuring progress

In order to achieve these desirable outcomes, define a strategy that accommodates change and which all of your stakeholders can follow.

Embracing the Rationale

The day you deliver the application is just the first day of its (hopefully) long and fruitful life. To lay the groundwork for success, a business analyst delivers not only the solution, but also a well-thought-out package of materials that his or her colleagues will use to maintain and enhance the application throughout its life.

The package includes not just requirements but also all of the analyses, discussions, debates, and trade-offs that led to the requirements. All of the reasoning and rationale that led to the requirement are often as important as the requirement itself.

Planning Your Integration with Development and Testing

Requirements are a means to an end. They are the blueprints (as it were) for the software solution that will be built and tested.

After your requirements are ready, the work is only beginning: your next job is to plan how your colleagues in development and quality assurance will consume this information efficiently and without misinterpretations.

Starting Small and Evolving Over Time



The requirements process, like any process, needs time to mature and develop. As such, don't attempt to cast in stone every facet of the requirements process right out of the gate. Allocate time to review what's working and what isn't and adjust the process with lessons drawn from each completed project.

This recommendation holds true when implementing a requirements tool as well: cover the basic requirement types and custom properties first and see what else is needed based on needs that arise in a given project. Adding new requirement types and properties to the tool is always easier than removing ones that really aren't needed but may already have information stored in them.



The worst approach you can take with your requirements process or tool is to overcomplicate it from the start as doing so hinders overall adoption and can quickly lead to the organization abandoning both the process and the tool.

WHAT NEXT?



Calculate your Requirements Definition and Management ROI

Use this simple tool and in 30 seconds see how much money your company can save by implementing a Requirements Definition and Management (RDM) tool like Blueprint.

Calculate your ROI at blueprintsys.com/lp/roi-calculator

Position Your Next Software Project for Success

Kick off your software project the right way by getting the best requirements product. Based on hundreds of customer implementations and industry best practices, this FREE RFP template will help you select the right modern Requirements Definition and Management (RDM) application for your business.

Get it now at blueprintsys.com/lp/rfp-template

toll free: 1-866-979-2583 | info@blueprintsys.com

These materials are the copyright of John Wiley & Sons, Inc.
and any dissemination, distribution, or unauthorized use is strictly prohibited.

SEE BLUEPRINT IN ACTION!

Join Us for Our Monthly Product Demo

See first-hand how Blueprint transforms the Business-IT relationship into a visual and engaging collaboration with less rework and reduced project cost resulting in on-time and on-budget applications.

Check available dates and times at
blueprintsys.com/lp/product-demo

toll free: 1-866-979-2583 | info@blueprintsys.com

These materials are the copyright of John Wiley & Sons, Inc.
and any dissemination, distribution, or unauthorized use is strictly prohibited.

Speed up your software development process with the help of this straightforward guide

Smart businesses know that high-quality requirements are the cornerstone of any successful software development project. This fun and friendly book is an introduction to the role that is central to requirements: the Business Analyst. It explains why the role is so critical and how Business Analysts are transforming software projects.

- *Get to know the challenges facing Business Analysts — discover the vital role they play in defining requirements and developing projects*
- *Help make your software projects more efficient — find out how Business Analysts can cut down on miscommunication, keep stakeholders engaged, and drive better results*
- *Explore how technology can help — learn all about technology that helps Business Analysts improve productivity and finish projects on time and on budget*



Open the book and find:

- Ways to better communicate and manage your requirements
- How to streamline your project's lifecycle
- Tips on how to maximize team efficiency and collaboration across the board
- Advice on how to choose the best requirements definition and management system

Making Everything Easier!™

Go to **Dummies.com**® for videos, step-by-step examples, how-to articles, or to shop!