

Mastering *the* Requirements Process

Build the Right Software – First Time

Requirements are the most misunderstood part of systems development, and yet the most crucial. Requirements must be correct if the rest of the development effort is to succeed. This workshop presents a complete process for eliciting the real requirements, testing them for correctness, and recording them clearly, comprehensibly and unambiguously.

Software development today has more demands on it than ever; and fewer resources to meet those demands. Getting the software right—the first time—is the most effective way to succeed under these circumstances. Today's requirements process is incremental with quick cycle times. It uses prototypes and scenarios, and it ensures that your developers know precisely what you—and your customer—mean when you write a fit criterion – a concise test case for the requirement.

This workshop shows you how to precisely define the scope of the business problem, to discover and involve the appropriate stakeholders, to use techniques such as apprenticing and use case workshops to learn what the users really need, to write testable requirements, and to phase the requirements to allow incremental delivery of the product.

You Will Learn How to:

- Determine your client's needs—exactly
- Write requirements that are complete, traceable, and testable
- Precisely define the scope of the project
- Discover the stakeholders and keep them involved
- Use up-to-date techniques such as storyboarding and e-collaboration
- Get the requirements quickly, and incrementally

Is This for Me?

Yes, if you want to be involved in delivering the right systems—the ones that get used. Your title is probably **business analyst, systems analyst, project leader or manager, requirements engineer, consultant** or similar. Or if you are a **user or software customer** and want to ensure the requirements process delivers what you need.

What Will I learn? What Will I be Better at?

- **Project Blastoff**

This builds a foundation for the requirements project by establishing its Scope-Stakeholder-Goals. This gives you the precise scope of the business area to be studied; a testable goal for the project; and using stakeholder maps, you can identify all the sources of requirements. Additionally, the blastoff ensures the project is viable and worthwhile.

- **Trawling for Requirements**

At the core of any requirements process is the ability to get people to tell you what they really need, rather than their perceived solution, or what they think you might be able to deliver. We show you how to use apprenticing, use case workshops, interviewing, brainstorming, mind maps and other techniques to discover exactly what the customers need—and want.

- **Functional Requirements**

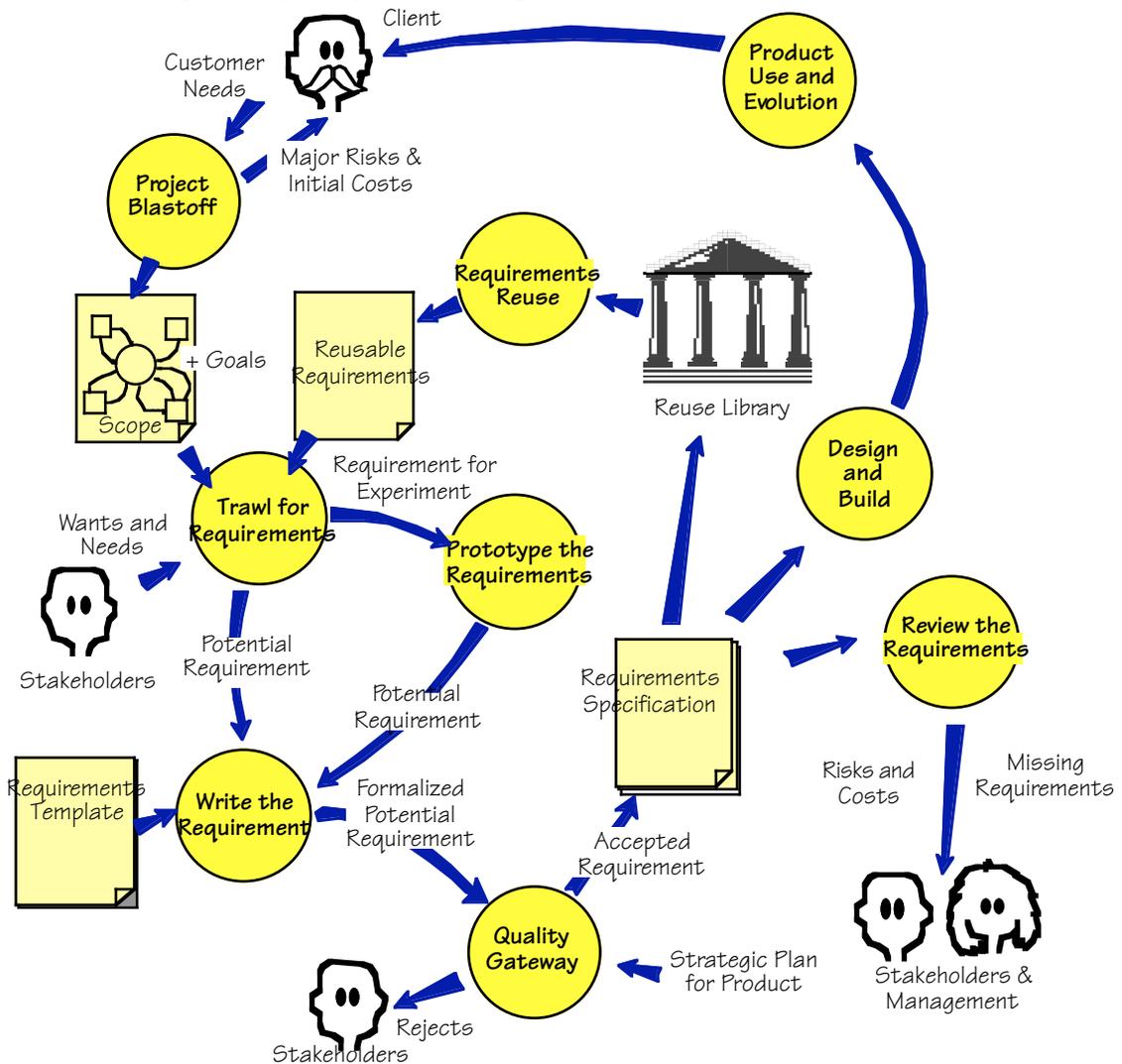
Functional requirements are those things the product must do. You discover them by understanding the work the user does, and determining what part of that work the automated product can best do. The resulting interaction between user and product is usually modeled with scenarios, and from these, you can readily derive the functional requirements.

- **Non-functional Requirements**

Non-functional requirements are properties the product must have, such as the desired look and feel, usability, performance, cultural aspects and so on. This section discusses the types of non-functional requirements, and shows you how to use the template, and other methods, to find the all-important qualitative requirements for your product.

- **Managing Your Requirements**

Requirements are the lynchpin of any development effort, and so have to be written correctly and managed effectively. This section demonstrates the use of a template to help you write requirements. It looks at requirements management issues like traceability, prioritization and conflicting requirements. We also look at tools to help manage requirements specifications.



- **The Quality Gateway**

Testing is most effective when it is done early in the development cycle. Here we demonstrate how to test requirements before they become part of the requirements specification. The Quality Gateway rejects out-of-scope, gold-plated, non-viable, incorrect and incomplete requirements. We show how you can attach an unambiguous *fit criterion* to a requirement. This makes the requirement testable, as well as ensuring the implemented solution precisely matches the customer's expectations.

- **Prototyping and Scenarios**

Some requirements are not discovered until the user has the opportunity to use the product. Prototyping is a way of discovering requirements by testing mock-up products for the user's work. Here we look at the merits of both low and high-fidelity prototypes, and how they and scenarios are used to discover previously-hidden requirements.

- **Your Requirements Process**

We look at how to make your own requirements process as effective and efficient as possible. For example, accelerating the requirements gathering by establishing the scope then building an early throwaway prototype before moving on to incremental delivery. Each part of the requirements process is examined so that participants can discuss problems and ideas related to their own situation, and how they can use the lessons from this course to improve their existing requirements process.

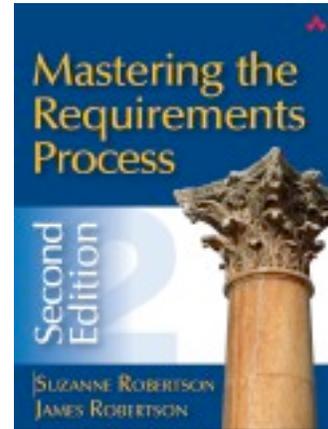
Workshops

We want you to use this right away. Each of the teaching chapters is reinforced with a workshop where you apply the concepts presented in the seminar. Participants work in teams to discover, specify and evaluate requirements for a significant system by:

- Defining the project's scope, its goals and the relevant stakeholders
- Identifying business use cases and product use cases
- Prototyping the product to find hidden requirements
- Applying the requirements specification template
- Defining functional and non-functional requirements
- Deriving the fit criterion, or measurement, for the requirements

There's More . . .

- Your instructor has real-world experience, and is able (and willing) to call on years of experience when discussing your particular requirements issues.
- Your own copy of the acclaimed *Mastering the Requirements Process*, Second Edition by Suzanne and James Robertson, published by Addison Wesley.
- A copy of the Volere Requirements Specification Template. This complete template provides the foundation for your own requirements specifications.
- A complete version of the Volere Requirements Process. This guides you through the intricacies of requirements gathering.
- Guidance on tools currently available to assist requirements capture and recording.
- References to books and sources of up-to-date requirements engineering techniques
- A final session where, through discussions, interaction and demonstrations, you can ensure you have a requirements process suitable for your organization.



Learning from Experience

Suzanne Robertson is co-author of *Mastering the Requirements Process, Second Edition* (Addison-Wesley 2006) a book that provides guidance on finding requirements and writing them so that all the stakeholders can understand them. Her other requirements book, *Requirements-Led Project Management* (Addison-Wesley 2005) addresses how to use requirements as input to planning and management. She is also co-author of the *Volere* approach to requirements engineering.

She has more than 30 years experience in systems specification and building. Her courses on requirements, systems analysis, design and problem solving are well known for their innovative workshops and practical applicability. Current work includes research and consulting on finding and involving the right stakeholders, the building of requirements knowledge models and running audits for assessing requirements specifications. She is a principal and founder of The Atlantic Systems Guild and is founding editor of the Requirements column in IEEE Software magazine.

James Robertson is a consultant, teacher, author, project leader whose area of concern is the requirements for products, and the contribution that good

requirements make to successful projects. James is a leading proponent of the principle of introducing creativity into the requirements process. His controversial article “Eureka: Why Analysts Should Invent Requirements” in *IEEE Software* has provoked heated discussion and has been widely quoted.

Before becoming a systems engineer, James trained as an architect and his experience in that profession provides inspiration for his work on innovation and creativity. He is co-author of *Mastering the Requirements Process, Second Editions* (Addison-Wesley 2006), *Requirements-Led Project Management* (Addison-Wesley 2005) and the *Volere* approach to requirements engineering. He is also a principal and founder of The Atlantic Systems Guild, a think tank known for its research into new systems engineering techniques.

Join the Move to Better Requirements

If you want to join the elite band of software developers whose systems are used—and enthusiastically used—then come and participate when one of the industry's most respected names explains how you can get the most value from your requirements gathering activities.

It will take three days out of your schedule, and we will give them back to you with interest (think how much extensive modification and abandoned systems cost you). We know that when you use a better requirements process, you will save months of maintenance effort, be more responsive to user requests, and avoid building systems that end up as shelfware.

Find out how you can gather requirements that deliver your systems earlier, and ensure that they are used, and useful.

How to Get This

Contact the Atlantic Systems Guild

London +44 (0)207 262 3395 or email james@systemsguild.com

New York +1 212 620 4282 or email tlister@systemsguild.com

Please also see our web sites www.systemsguild.com and www.volere.co.uk