

Software Development
[August 2002](#)

Come Together

Using Jerry Weinberg's classics as a springboard, facilitator Ellen Gottesdiener's new book, *Requirements by Collaboration*, offers a fresh look at the necessities of getting along.

By [Warren Keuffel](#)

If there's anyone I'd consider a guru, it would have to be Jerry Weinberg. Weinberg's prolific oeuvre includes such classics as *An Introduction to General Systems Thinking* (Wiley, 1975; Silver anniversary edition, Dorset House, 2001) and *The Psychology of Computer Programming* (Van Nostrand Reinhold, 1971; Silver anniversary edition, Dorset House, 1998). But the Weinberg title I return to time and time again is *Exploring Requirements: Quality Before Design* (Dorset House, 1989), which Weinberg coauthored with Don Gause.

Though it may be difficult for some readers to imagine that a book written back in the 20th century could remain relevant today, this one does. Weinberg has the unique ability to make you expand your critical thinking. In *Exploring Requirements*, he tackles resolving ambiguity in human communication as it relates to elucidating a project's real design requirements.

Note that the project need not be a *software* project—the same obstacles face any group developing any kind of product. But, as it happens, Weinberg's audience tends to be primarily software developers, and software developers are frequently charged with reconciling the demands of multiple stakeholders. In such situations, it's easy to fall into the trap of treating all requirement demands as if they were created equal.

Finding Efficiency

Exploring Requirements is not a technical book, but rather a unique melding of systems analysis techniques with guidelines for channeling group psychology into productive directions. This latter activity describes the role of the facilitator—an individual who helps a group achieve its goals with maximum efficiency.

Ellen Gottesdiener, a noted facilitator who lectures frequently at Software Development conferences, has recently authored *Requirements by Collaboration: Workshops for*

Defining Needs (Addison-Wesley, 2002), which helps provide a more defined structure to put the concepts of *Exploring Requirements* to use.

According to Gottesdiener, a requirements workshop—as opposed to an ordinary meeting—is a "structured meeting in which a carefully selected group of stakeholders and content experts work together to define, create, refine and reach closure on deliverables (such as models and requirements) that represent user requirements."

The People Factor

But Gottesdiener places as much—and perhaps more—emphasis on collaboration as on deliverables. As she told me recently, "As facilitator, it's very exciting and gratifying to see people connect and to be the yeast that helps collaboration grow."

The results of that collaboration are ideally found in the workshop deliverables, which should answer the questions of who, what, when, why, where and how (see sidebar). This enables Gottesdiener to go beyond a single-focus deliverable (such as UML) and incorporate results from other disciplines.

If Gottesdiener's book was merely a laundry list of requirements deliverables, it'd be indistinguishable from the multitudes of other works about requirements. But she devotes a great deal of attention to helping her readers run successful workshops, describing how to channel team energy into useful work toward a shared purpose. Of particular interest is her discussion of the various forms of decision-making, and knowing when to utilize each. If you'd like to sample Gottesdiener's writing about collaboration, her article "[Decide How to Decide](#)" (Jan. 2001) may be of interest.

Write me directly at wkeuffel@acm.org, or post a message for all to read at the [Interface forum](#) at Sdmagazine.com.

Gottesdiener's Six Great Focus Questions

Questions	Deliverables
Who?	Stakeholder Classes, Actor Map, Actor Table, User Interface Navigation Diagram, Prototype
What?	Relationship Map, Glossary, Context Diagram, Domain Model
When?	Event Table, Statechart Diagram
Where?	Locations, Platforms (hardware and software)
Why?	Business Policies, Business Rules, Design Table or Tree
How?	Process Map, Use Cases, Scenarios, Use Case Map, Use Case Package