



November 4, 2002

# Do the Math: Strong Requirements Practices Save Spiraling Project Costs

*Margo Visitacion*

## Catalyst

A client inquiry

## Question

What cost/benefit analysis information (business case justification) can Giga provide for implementation of requirements management?

## Answer

Difficulty in justifying requirements management (RM) continues to be a surprising trend in software development projects, despite consistently high failure rates. The numbers alone should support improving requirements management practices. More than 60 percent of all software projects in the United States fail and poor requirements remain one of the top five reasons. A 2001 study showed that 40 percent of projects fail to meet business objectives within one year of implementation and it was estimated that the US wasted more than \$78 billion on projects that were either unsuccessful or cancelled. Despite the compelling numbers, companies continue to resist implementing requirements management. Companies struggling with project delivery should analyze the hidden costs surrounding development efforts.

According to *Software Engineering Economics* (Barry Boehm, Prentice Hall PTR, 1981), requirements efforts usually encompass 8 percent to 15 percent of a project, but can be traced as the source of 85 percent of the defects. Projects that have no formal requirements management practices have more than 50 percent more defects than projects with sufficient RM practices. There are a few reasons for this; because of the flexible nature of software development, companies don't see the tangible benefits up front and, therefore, don't feel the need to employ formal RM practices. Efforts that do spring up are often ill-defined and emphasize one area (e.g., user requirements) over another (e.g., functional requirements), or companies become so dependent on prototyping to draw out business involvement, they become overly concerned with user experience to the detriment of designing robust systems. Companies wanting to improve delivery and quality within software projects should do the math. Previous studies find that a bug found post production or in the final certification process of a project takes approximately eight hours to fix, while one found in the requirements or inspection phase takes an average of 15 minutes. Planning for and executing requirements management practices allows project teams to prevent unnecessary cost overruns.

Requirements management tools increase the probability of finding potential defects earlier in the cycle. By employing RM tools, organizations are able to centralize all requirements information, manage changes and alert teams of change through requirements traceability functionality that highlight and notify users of suspect links (e.g., when a requirement has changed and was not updated in all relating documentation). Using robust RM practices wrapped around RM tools, companies reduce rework by validating requirements earlier, finding and repairing potential defects earlier (see Planning Assumption, [Update: Implementing Requirements Management](#), Margo Visitacion).

Consider the table below, which is an example of an average project team with 10 members at \$100 per hour, three resources allocated (developer, database analyst and quality assurance analyst) to resolve three defects found late in the cycle. Each defect takes eight work hours per person to repair (one day to debug and code, one day to integrate and adjust DBMS fields and one day to test and validate).

### Cost of Defects Found per Phase

Total number of defects found	Phase found	Work hours per defect per person	Resources allocated	Total work hours per defect	Total work hours * hourly rate (\$100/hr) per defect	Total cost for all defects found
3	Certification	8	3	24	\$2,400	\$7,200
3	Requirements Inspection	.15	3	.45	\$45	\$135
3	Unit Testing	.30	3	.90	\$90	\$270

Source: Giga Information Group