adapting agility . . . .

Project Management in Plan-Based and Agile Companies

Martina Ceschi, Alberto Sillitti, and Giancarlo Succi, *Free University of Bolzano-Bozen*

Stefano De Panfilis, *Engineering Ingegneria Informatica*

> Survey results show that adopting agile methods appears to improve management of the development process and customer relationships.

Surveys of more than 8,000 projects show that most project failures involve stakeholder problems.[1] Notably, five of the top six reasons for failure stem from communication problems between the development team and the customer.[2] In 2000, the Standish Group identified 10 factors for project success. Of these, the second and third most important were "user involvement" and an "experienced project manager."[3] From this you can conclude that most projects fail because of people and project management issues rather than technical issues.[4]

Agile methods (see the related sidebar on page 22) are a recent set of development techniques that apply a human-centered approach to software production.[5] The agile approach aims to deliver high-quality products faster, producing satisfied customers.

We conducted an empirical study to investigate whether agile methods change and improve project management practices in software companies. We based the study on Henry Petroski's approach, which says that the "analysis of causes of failures can do more to advance knowledge than all the successes in the world."[6] The underlying data sample comprises managers of software companies—10 adopting agile methods and 10 using traditional (which, for simplicity, we call plan-based) approaches.

Structure of the investigation

Our study focuses on how well project management deals with people (developers and customers) and the development process (planning and organization). We based the study's overall structure on Victor Basili's Goal/Question/Metric approach.[7]

GQM aims to systematically develop a goal-oriented measurement program. The measurement system has three levels:

- *Conceptual*: goal;
- *Operational*: question; and
- *Quantitative*: metric.

GQM defines a goal, refines this goal into questions, and defines metrics providing information to answer these questions.

In our study, our goal was to monitor what a project manager of a company (agile or plan-

Agile Methods

Lean production is a set of practices focused on the continuous improvement of the production process, by identifying and removing anything that doesn't add value to the customer.[1] It was born in the manufacturing environment, but many basic principles also fit the software industry.

Agile methods apply the principles of lean production to the overall software life cycle.[2] So, these methods focus on providing value for the customer and support requirements variability, but they don't fit every application domain.

The agile approach is more people-oriented rather than process-oriented.[3,4] This means that it depends heavily on individual skills. Agile methods claim that people make projects successful and that no process will ever make up for the lack of the development team's skill; so, a process's role is to support the development team.[5] Moreover, these methods promote the cohesion of team members and developer and customer interaction.

Two popular agile techniques are Extreme Programming and Scrum. XP is a collection of principles and practices that aims at enabling successful software development despite vague or constantly changing requirements in small- and medium-size teams.[6] Scrum aims to manage the development process through an empirical approach that applies the ideas of industrial process control theory to software development.[6]

References

1. T. Ohno, *Toyota Production System: Beyond Large-Scale Production*, Productivity Press, 1988.
2. T. Poppendieck and M. Poppendieck, *Lean Software Development: An Agile Toolkit for Software Development Managers*, Addison-Wesley, 2003.
3. K. Beck et al., "Manifesto for Agile Software Development," 2001, http://agilemanifesto.org.
4. K. Beck, *Extreme Programming Explained*, Addison-Wesley, 2000.
5. J. Highsmith and A. Cockburn, "Agile Software Development: The People Factor," *Computer*, Nov. 2001, pp. 131–133.
6. P. Abrahamsson et al., *Agile Software Development Methods*, VTT, 2002; www.inf.vtt.fi/pdf/publications/2002/P478.pdf.

based) considers important to improve process, organize teams, and solve problems. Our context was that of managers in local and international software companies.

We had several questions to consider—namely, determining the main problems in the development process and solutions adopted, evaluating process planning and organization, estimating the relationship with the customer, and assessing the managers' actual knowledge (benefits and disadvantages) of agile methods. We used a phone questionnaire to quantitatively measure the answers to these questions.

The questionnaire

The nature of questionnaires leaves them open to information loss and lack of integrity of the collected data.[8] One way to address this problem is to collect a massive number of questionnaires.[9] However, this approach doesn't work well in the case of software managers because they're typically very busy and it's difficult to obtain their time for these studies.

To address these issues, we took extra care in iteratively defining the questionnaire and in collecting the data. We ensured that each question, the questionnaire, and the responses adhered to Giorgio Marbach's soundness principles.[8] The questionnaire, as a whole, also followed Jean Converse and Stanley Presser's psychological criteria on question order and on avoiding critical terminology.[9]

The questionnaire comprises 25 questions grouped in four sections:

■ the company and the interviewee's status, primary software development problems, and adopted solutions;
■ the development process's planning and organization;
■ the relationship with the customer and characteristics of good developers;
■ the managers' knowledge of agile methods, their actual use, and the advantages or disadvantages of their use.

We began our process for finalizing the questionnaire and collecting data by asking a group of students to evaluate the first draft, and six managers from different companies to evaluate the second. Next, we selected 20 respondents from companies with some correspondence to the authors' primary research—10 working in agile companies and 10 in plan-based companies. We emailed the final questionnaire to the managers and asked them to read it carefully. We then collected the questionnaire responses through phone interviews, which occurred typically one week after our email. We recorded the interview results on paper and asked the interviewees to check them by email. Only when we received positive confirmation of the responses did we accept the questionnaire.

Survey sample

The 20 interviewees are project managers in software companies—19 males and one female—and the average age is 40. Eighteen managers have a university degree and two have completed secondary school. Seventeen companies, more than 80 percent, were founded after the '80s. Fourteen of the selected companies are located in Italy, five in the US, and one in Switzerland.

The sample contained 10 companies that defined themselves as agile and 10 that didn't. The companies' business areas differ—for example, consulting, services, and software development. Most have a large number of employees, and only three have fewer than 10 employees. All the companies were exposed to production process changes over the years. The main causes for such changes included changes in customers' requirements (50 percent), technology changes (70 percent), and other reasons (40 percent) such as failure with the prior development process, the wish to improve the process's efficiency and effectiveness, or the adoption of quality systems (such as ISO 9001).

We can't consider this sample as representative of the entire population of software companies. Nevertheless, the agile and plan-based companies selected share similar characteristics: the companies' years in the market, business areas, and number of employees, and the respondents' roles, gender, age, and education.

Results analysis

The number of managers our study considered is small, and it's not possible to perform a complex statistical analysis. Therefore, we based our analysis on the comparison of percentages.

Primary problems

According to 15 managers, the main problem in software development is delivering products with all the features on time. The Standish Group's 2000 Chaos survey of 8,000 projects obtained a similar result: only 26 percent of the projects were completed on time, on budget, and with all the originally planned functions.[3]

Approximately 50 percent of the plan-based companies and 10 percent of agile companies believe they have a difficult relationship with their customers (see Figure 1a).

However, agile companies' customer relationships are not so difficult to manage. In fact, one of the main problems solved by adopting agile methods is the customer relationship (see Figure 1b).

The selected managers adopted different, sometimes multiple, solutions for delivering software with all the features on time: using new development methods such as Extreme Programming (XP) and Scrum (half the managers), improving productivity processes (more than half the managers), and focusing on people (customers and developers) by improving
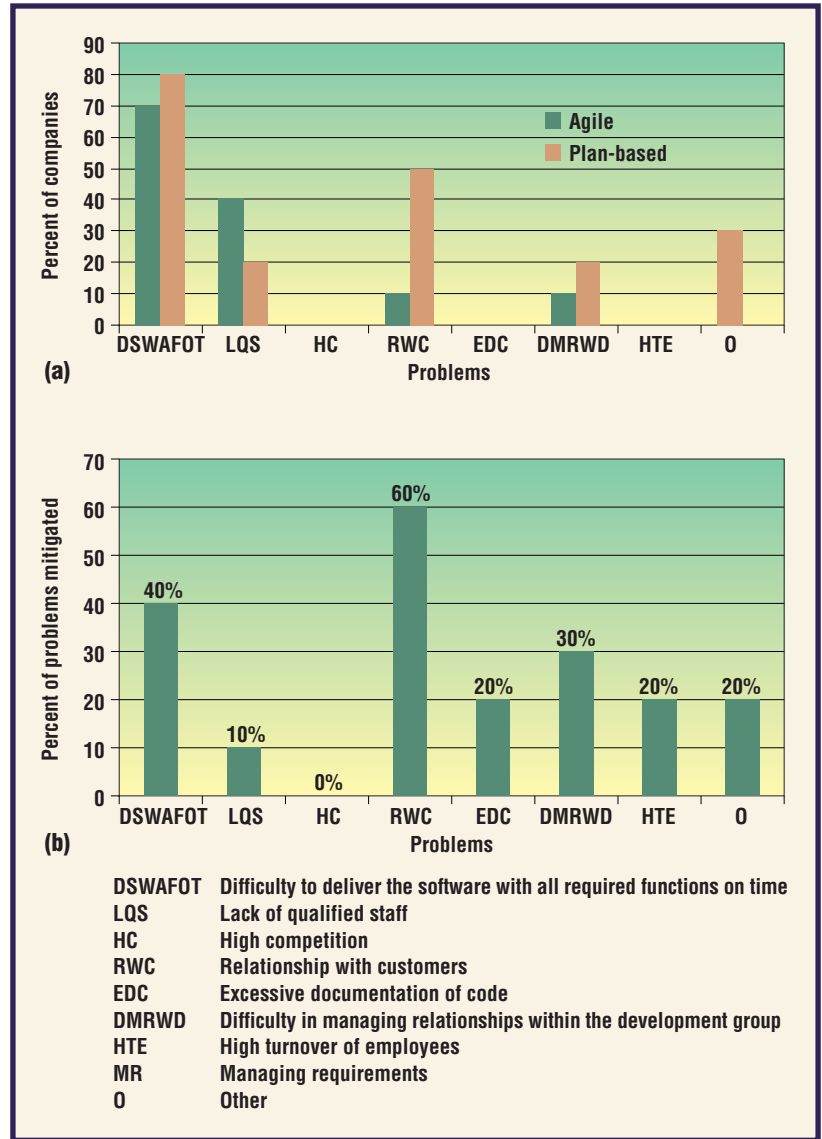
communication and knowledge transfer such as small size and good communication between customers and developers (30 percent of the managers).

Planning and organization

Most of the considered agile and plan-based companies (70 percent) include in their process incremental code development (see Figure 2a). Half of the plan-based companies also split the process into tasks, and 40 percent split them into phases. Moreover, a considerable number of companies use prototypes before developing the final product.

Agile companies prefer to organize their processes in more releases and pay attention to activity planning. Most agile companies tend to
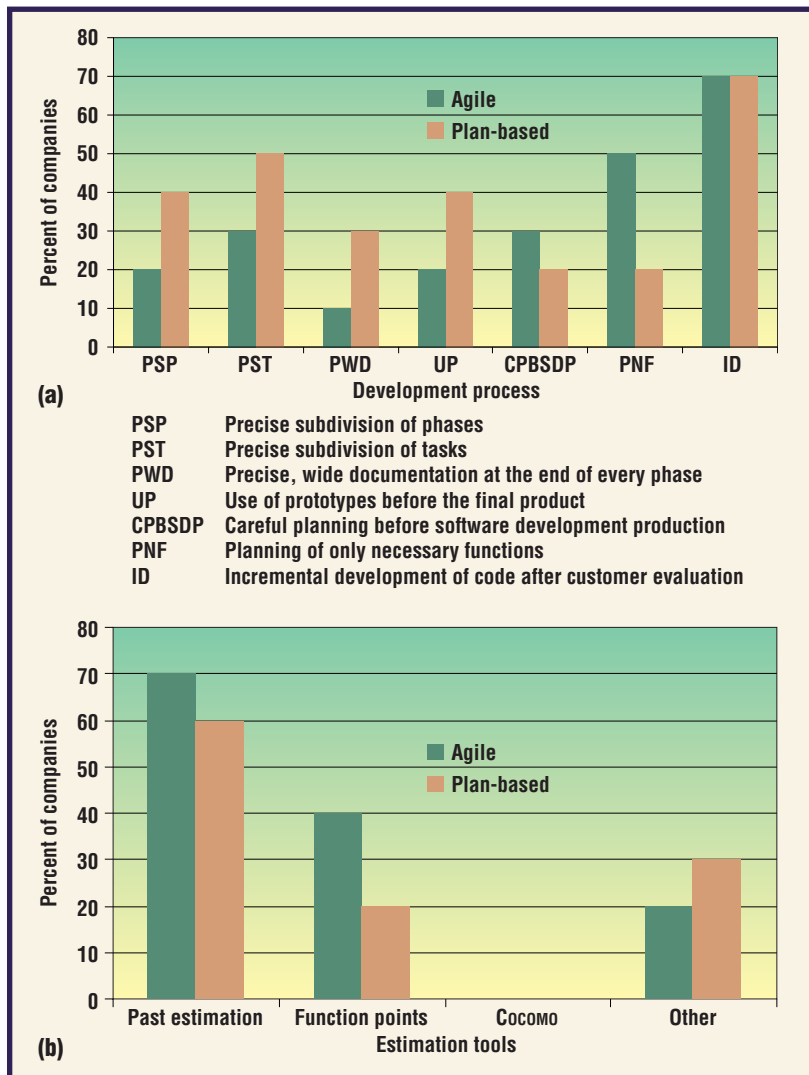

Figure 1. (a) Primary software development problems and (b) software development problems mitigated by adopting agile methods.

DSWAFOT	Difficulty to deliver the software with all required functions on time
LQS	Lack of qualified staff
HC	High competition
RWC	Relationship with customers
EDC	Excessive documentation of code
DMRWD	Difficulty in managing relationships within the development group
HTE	High turnover of employees
MR	Managing requirements
O	Other

(a)

Abbreviation	Description
PSP	Precise subdivision of phases
PST	Precise subdivision of tasks
PWD	Precise, wide documentation at the end of every phase
UP	Use of prototypes before the final product
CPBSDP	Careful planning before software development production
PNF	Planning of only necessary functions
ID	Incremental development of code after customer evaluation

(b)

Figure 2. Process management: (a) how firms organize and plan their development process and (b) the estimation tools used in agile and plan-based companies.

	Table 1	
Managers' satisfaction with project planning		
Satisfaction	**Agile (%)**	**Plan-based (%)**
Not at all	0	20
Not very	0	0
Sufficiently	80	60
Very	20	20

plan and, consequently, to develop only essential functionality at each iteration. However, this doesn't mean they don't carefully plan their development processes. In fact, they're more satisfied with the way they plan their projects than plan-based companies are (see Table 1).

Eight-five percent of the managers would like to improve process planning, even though 70 percent of managers are sufficiently satisfied with it and 20 percent are very satisfied.

It's interesting how companies use common planning and organization tools to improve the development process. These tools include Gantt charts (65 percent), PERT (Program Evaluation Review Technique) charts (40 percent), the critical path method (20 percent), and others (30 percent).

The main method that agile and plan-based companies use for the feasibility analysis and for planning is based on past estimations and experiences (see Figure 2b). However, it's difficult to have enough homogeneous projects over the years in order to calculate reliable estimates; requirements, development teams, technologies, and customer needs change in every project, and most developers don't develop exactly the same product twice.

Customer relationships

Both agile and plan-based companies have collaborative customer relationships. However, the way they manage them differs. Sixty percent of the agile companies have their customers on-site, meaning that the customer is directly available for most of the project. A smaller number of plan-based firms (40 percent) use this practice as well.

Understanding what customers really want requires their constant involvement in the project. This participation implies interaction between the customer and the development team to allow quick response to changes. In this context, it's difficult to keep the contract's variables (scope, price, and time) fixed over time. Thus, agile companies tend to regulate their customer relationships with flexible contracts instead of fixed ones that predefine functionalities, price, and time (see Figure 3a).

Increased customer contact creates a high-quality link between the development team and the customer. Consequently, agile companies are more satisfied with their customer relationships than plan-based companies (see Figure 3b).

Agile and plan-based companies disagree on what constitutes the most important customer problems. Most plan-based companies (70 percent) consider changing requirements one of their more critical issues (see Figure 3c).

It's difficult for a company with a traditional development process to respond rapidly to unexpected changes, especially if the implementation phase has already started. Every modification runs the risk of affecting the project plan and the process organization.

Typical solutions for coping with changing requirements include

- attempting to anticipate the requirements that could change over time during the analysis phase;
- creating an initial, detailed requirements specification through formal documents;
- applying more constraints on the contracts; and
- simply attempting to please the customer and complying with requests.

However, all such solutions are not satisfactory.

Agile companies worry less about variations in requirements. They use an iterative process, during which the customer may refine and modify requirements. Moreover, having the customer on-site encourages requirements changes.

Typically, the surveyed plan-based companies deliver the entire product in one batch at the end of development, and customers can evaluate the system only at the end. Consequently, their customers frequently demand to speed up the project, a common and substantial problem. Often, these customers don't understand the actual complexity of implementing their requests.

To address this problem, plan-based companies often

- develop the product with parallel teams,
- communicate with customers to explain the problems of delivering the product too quickly, and
- use prototyping and allocate additional time in the budget as a precaution.

Agile companies encounter the request to deliver the final product faster less often because of the frequent product releases during development. Incrementally delivering functionality in a sequence of releases appears to better satisfy customer needs.

Qualities of good developers

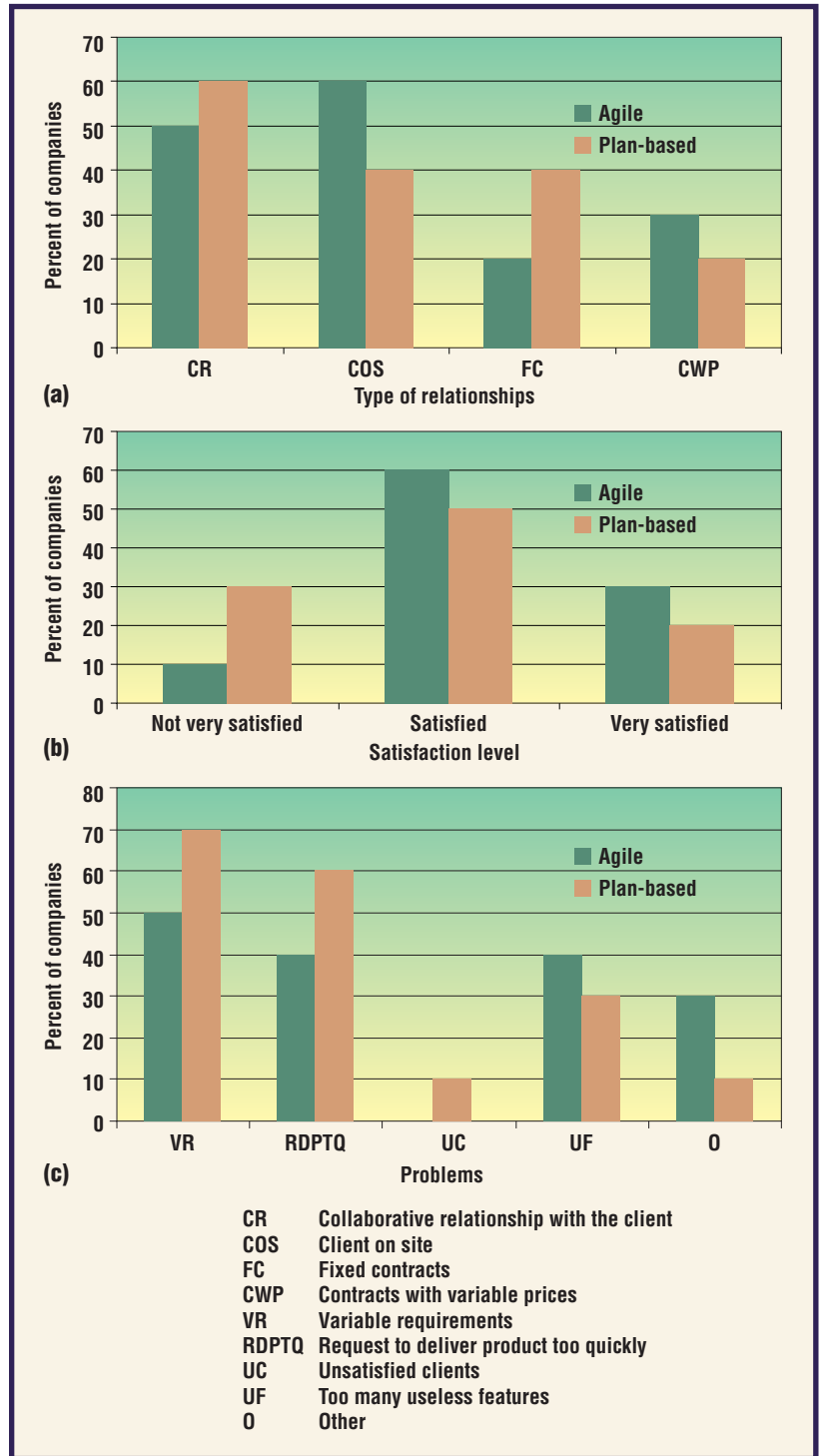Both agile and plan-based companies have similar responses about what qualities good developers have. In both cases, they highlight the ability to work in groups, with a clear preference for developers who can work in teams (50 percent of the answers) over developers with high individual ability (20 percent) or who are strongly motivated (30 percent).

Some patterns exist in the answers: nearly all the companies that consider teamwork as the

CR   Collaborative relationship with the client
COS   Client on site
FC   Fixed contracts
CWP   Contracts with variable prices
VR   Variable requirements
RDPTQ Request to deliver product too quickly
UC   Unsatisfied clients
UF   Too many useless features
O   Other

Figure 3. The customer relationship: (a) types of relationships, (b) satisfaction level, and (c) typical problems.

Table 2
Summary of results

Survey questions	Agile companies	Plan-based companies
Why have you modified your development process?	Changes in the customers' requirements, 50% Changes in the technology, 70% Other reasons, 40%	Changes in the customers' requirements, 50% Changes in the technology, 70% Other reasons, 40%
Which is your most difficult software development problem?	Delivering the software with all the features on time, 70% Customer relationship, 10%	Delivering the software with all the features on time, 80% Customer relationship, 50%
Which software development problems have you solved by adopting agile methods?	Delivering the software with all the features on time, 40% Customer relationship, 60%	— —
How do you plan and organize the software process?	Incremental code development, 70% Precise subdivision of tasks, 30% Precise subdivision of phases, 20% Use of prototypes, 20% Planning only essential functions, 50%	Incremental code development, 70% Precise subdivision of tasks, 40% Precise subdivision of phases, 50% Use of prototypes, 40% Planning only essential functions, 20%
How do you consider your relationship with your customer?	Not very satisfied, 10% Satisfied, 60% Very satisfied, 30%	Not very satisfied, 30% Satisfied, 50% Very satisfied, 20%
What are the most important problems that you usually face with customers?	Variable requirements, 50% Requests to deliver the product too quickly, 40%	Variable requirements, 70% Requests to deliver the product too quickly, 60%
What is a developer's most important quality?	Ability to work in a group, 50% High individual ability, 20% Motivation, 30%	Ability to work in a group, 50% High individual ability, 20% Motivation, 30%

most important quality consider motivation second most important and high individual ability third. Moreover, all respondents who said that motivation is the most important quality consider the ability to work in groups second and high individual ability the least important. Thus, all respondents show an interesting regularity in their opinions on this topic.

The interviewed managers adopted several solutions to improve these three developer qualities: continuous training improves individual abilities, and regular communication and project involvement improves teamwork ability and motivation.

Agile methods

The last part of the questionnaire deals with the manager's knowledge of agile methods and of their advantages and disadvantages. About 90 percent of the plan-based company managers know about agile methods even if they don't use them. This means that these approaches are a well-known phenomenon.

The main causes for nonadoption of agile methods are superficial knowledge of the topic, resistance inside the company and from customers, and large or geographically separated teams. These results correspond to XP's limits as pointed out by Kent Beck.[5]

A perfect correspondence exists between knowledge of agile methods and knowledge of XP. All the plan-based and agile companies know XP, but only 60 percent of the agile companies and none of the plan-based ones know about Scrum.

Regarding the advantages of the use of agile methods, recall the correlation between software development's main problems and the problems that agile methods address (see Figure 2). Adopting the agile approach offers a good solution for the top two software problems.

According to our study, introducing agile methods also offers improvements in quality, requirements management, customer satisfaction, and team satisfaction. The main problems caused introducing agile methods are the lack of a detailed preliminary cost evaluation and the troubles new concepts (such as pair programming, test first, and customers on-site) cause. The primary difficulty seems to be cultural: people (customers and developers) don't easily accept drastic changes in traditional environments.

Results summary

Table 2 summarizes this study's main questions and answers. In addition to the conclusions reported throughout the article, we further expand a few observations here.

We noted that while environment variables (both requirements and technology) affect all the companies, the agile ones can better protect the customer from most of the negative effects.

Additionally, almost all the plan-based companies say they know agile methods. However, many managers only have a superficial knowledge of them and are not aware that agile methods comprise a set of development methods and not a single one. This limited knowledge contributes to the nonadoption of such techniques; however, large or geographically separated development teams are often the primary cause.

This study's main limitation is the small sample size, which results from the investigation's structure of addressing top managers. The total number of top managers is limited, and we only interviewed one per company. It was difficult to persuade these managers to spend time in an interview; however, this difficulty makes the study more interesting because it includes valuable information collected directly from top decision makers.

This article has given a first analysis of the advantages and disadvantages of adopting agile methods from a project management perspective.

Because sample size limited the study, we plan to expand our data with input from other top managers. 🅜

About the Authors

Martina Ceschi is a research assistant and PhD student at the Faculty of Computer Science at the Free University of Bozen, Italy. Her research interests include agile project management and requirements management. She received a Laurea in economics from the University of Trento. Contact her at the Faculty of Computer Science, Free Univ. of Bozen, Piazza Domenicani, 3, I-39100, Bolzano, Italy; martina.ceschi@unibz.it.

Alberto Sillitti is an assistant professor at the Free University of Bozen, Italy. His research interests include software engineering, component-based software engineering, Web Services integration and measurement, agile methods, and open source. He received his PhD in electrical and computer engineering from the University of Genoa and is a professional engineer. Contact him at the Faculty of Computer Science, Free Univ. of Bozen, Piazza Domenicani, 3, I-39100 Bolzano, Italy; alberto.sillitti@unibz.it.
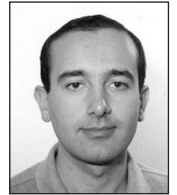
Giancarlo Succi is a professor of software engineering and the director of the Center for Applied Software Engineering at the Free University of Bozen. His research interests include agile methodologies, open source development, empirical software engineering, software product lines, software reuse, and software engineering over the Internet. He received a PhD in electronic and computer engineering from the University of Genoa. Contact him at the Faculty of Computer Science, Free Univ. of Bozen, Piazza Domenicani, 3, I-39100 Bolzano, Italy; giancarlo.succi@unibz.it.

Stefano De Panfilis is the director of the Research and Development Laboratory of Gruppo Engineering. His research interests include software development and process customization, support, monitoring, and control. He received a Laurea in mathematics from the University of Rome - La Sapienza. Contact him at the Engineering Ingegneria Informatica S.p.a., Via San Martino della Battaglia 56, I-00185 Roma, Italy; depa@eng.it.
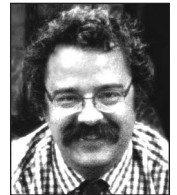
References

1. J. Johnson, "Turning Chaos into Success," *Software Magazine*, Dec. 1999, p. 30.

2. B. Boehm, "Software Engineering Is a Value-Based Contact Sport," *IEEE Software*, Sept./Oct. 2002, p. 97.

3. "Extreme Chaos," The Standish Group, 2001, www.standishgroup.com/sample_research/PDFpages/extreme_chaos.pdf, p. 4.

4. R. Thomsett, *Third Wave Project Management*, Yourdon Press, 1993.

5. K. Beck et al., "Manifesto for Agile Software Development," 2001, http://agilemanifesto.org.

6. H. Petroski, *To Engineer Is Human: The Role of Failure in Successful Design*, Vintage Books, 1992.

7. V.R. Basili, *Software Modeling and Measurement: The Goal/Question/Metric Paradigm*, tech. report CS-TR-2956, Dept. Computer Science, Univ. Maryland, College Park, 1992.

8. G. Marbach, *Le Ricerche di Mercato*, Utet, 1996.

9. J.M. Converse and S. Presser, *Survey Questions: Handcrafting the Standardized Questionnaire*, Sage, 1986.

For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/publications/dlib.