

# The Role of Plan-Based Approaches in Organizing Agile Companies

Alberto Sillitti, Giancarlo Succi  
Free University of Bozen  
{Alberto.Sillitti, Giancarlo.Succi}@unibz.it

## Abstract

*The paper presents the results of a survey performed in 23 Agile companies. The survey focuses on the identification of the non-agile practices used in the organization of their development process. Even if the companies considered in the study are using Agile approaches to develop their products, the organization and the planning of the work is deeply affected by the approaches used in Plan-Based companies. Understanding which are the most used non-agile techniques and why managers are still so linked to them can identify potential weaknesses in the actual implementation of the Agile Methods.*

## 1. Introduction

In the last few years more and more companies are experimenting Agile Methods (AMs) [5] to verify whether they are able to improve the effectiveness of software development. Among the main aims of AMs there are:

1. effective management of unstable requirements
2. frequent delivery of working software with valuable functionalities for the customer
3. reduction of useless documentation (this does not mean the reduction of the useful one)

Such goals are achieved reorganizing completely the development process, the structure of the development teams, the interactions among team members and with the customers, etc. This radical change does not only affect people developing software and their managers but also middle managers and top executives that are not directly involved in the development [17].

In particular, middle managers and top executives are used to ask detailed information to development teams to plan, allocate and forecast budgets, make strategic decisions, etc. In Plan-Based approaches, such information is delivered in very different ways, including: reports on the status of a project, financial reports, Gantt and Pert charts, etc. Due to the high level of uncertainty in software projects, everybody knows that such documents are often affected by large errors that make them unreliable. However, they are the only widely available tools to make high-level decisions [14].

In AMs, the production of such documents is considered a waste of time since they are unreliable and do not provide any value to the customer. Therefore, AMs do not include the production of such documents in the development process [1].

This lack of documents is often a problem for managers: there is a real need to understand the achievements of the development teams and make correct strategic decisions. The difficulty to get such information from Agile teams is often perceived as an impediment for implementing AMs [9, 10, 11, 12].

According to many middle and top executives interviewed in this study that have adopted AMs, AMs do not include sufficient tools to provide reliable and useful information to them. Using AMs, they would like to improve not only the development process but also their ability to make reliable estimates and correct strategic decisions. Due to the lack of specific tools for collecting and communicating such information in AMs, many Agile teams have borrowed tools from the Plan-Based approaches and they are trying to make them working in such new environments.

To investigate this aspect, we have used empirical data obtained in a survey conducted in 2004. This survey addressed most of the issues about project management, including the usage of Plan-Based approaches inside Agile teams.

This paper is organized as follows: section 2 presents the background of the study; section 3 describes the structure of the investigation; section 4 presents the results; finally, section 5 draws the conclusions.

## 2. Background

### 2.1 Agile Methods

Agile Methods are a set of development and management techniques designed to address the increasing uncertainty and complexity of software development.

Most of the key principles and practices of AMs comes from well-known theories in manufacturing. In fact, AMs incorporate techniques from Theory of Constraints (1984), Just-In-Time ('70s), Total Quality Management ('80s), and Lean Production (end of '80s) [2].

The main differences between Agile and Plan-based approaches are [16]:

- *AMs are adaptive rather than predictive.* AMs are designed to work in unstable and uncertain environments, and when the application domain is not well understood.
- *AMs are people-oriented rather than process-oriented.* They rely on people's expertise and competency. They focus on direct, frequent, team collaboration and communication.

The main values of AMs are [5]:

- *Individuals and interactions* over processes and tools
- *Working software* over comprehensive documentation
- *Customer collaboration* over contract negotiation
- *Responding to change* over following a plan.

AMs consider more important the left side of the Agile Manifesto value statements. Nevertheless, this does not mean that the items on the right are not important. They try to find a balance between the two sides, keeping the organization adaptive, people-oriented and focused on the customer [9].

The extraction and the communication of information regarding the status of a project to fulfill the needs of the middle and top managers are not included in the development process even if many AMs provide tools to access the status of the project from the people directly involved in the development process.

There are several AMs available, including [1]: Extreme Programming (XP), SCRUM, Adaptive Software Development (ASD), Crystal Methods, Feature Driven Development (FDD), etc.

Every Agile Method defines the structure of the development process in fairly different ways. Nevertheless, all of them are based on the key values listed above.

## 2.2 Project management

Plan-Based methods focus on standardized and well-defined processes that are used to make the activities predicable and repeatable. The basic assumption comes from the manufacturing mindset: through specialization it is possible to achieve a higher efficiency in the development process [15].

In Plan-Based organizations, people have well defined roles, responsibilities, and follow detailed plans and predefined workflows [6]. They also produce several documents related to the status of the project, forecasts, charts, etc. that are used by middle and top managers to make strategic decisions.

The team structure of the Plan-Based companies can produce some negative effects:

- Role-based teams tend to lead to islands of knowledge, difficult to share [7].
- Documentation might not be an effective mechanism for transferring tacit knowledge [9].
- The separation of the roles (or functions) could inhibit the communication and the collaboration across different teams [3].

AMs adopt an incremental development process. The main activities are performed in parallel and there is not a well-defined separation among project phases [10]. AMs emphasize the human role in software development, as opposed to institutionalized processes and development tools.

Due to the organization of the process, AMs promote the use of cross-functional teams. Such teams put together individuals able to perform all roles. Rotations from one role to another are common practices in Agile organizations [7]. Moreover, AMs advocate a different management approach, based on the idea of leadership-collaboration and self-organizing teams [12].

The managers set goals and constraints, providing boundaries within which the members of the development team can make decisions and assume responsibilities for those decisions. Self-organizing teams are not leaderless teams; they are teams that can reorganize themselves in various configurations to meet challenges as they arise [12]. In this flexible structure, the information flows also in a horizontal way.

AMs facilitate knowledge sharing (especially tacit knowledge sharing) and coordination primarily through face-to-face communication and collaboration. Relying on interactions between individuals facilitates sharing information and changing the process quickly when it needs changing. Frequent interaction between individuals compensates for minimizing documentation [11].

Agile company organization and team structure can have some disadvantages:

- Rotation of roles requires highly skilled and experienced people [12].
- Knowledge sharing through communication requires a high level of mutual trust among team members and frequent interactions [9].
- Middle and top managers are not involved in the direct communication. This results in a difficulty to collect information from Agile teams and make good strategic decisions [17].

### 3. The survey

In 2004 we conducted an empirical survey on project management in software development. We investigated different areas of project management including the usage of Plan-Based project management tools in Agile teams. We interviewed 23 managers of software companies that defined themselves as agile.

#### 3.1 Design of the Survey

The design of the survey follows the assumptions of Silverman [19] and the principles of the GQM approach [4]. We have chosen to perform a qualitative survey, using a semi-structured questionnaire as research methodology.

Given the nominal nature of most variables, the characteristics of the questions, and the low number of data gathered, we have not used the existing non-parametric statistical tests (e.g., binomial test and chi-square test). The statistics used are mode, frequency count, and the relative frequency distribution [18].

The final form of the questionnaire was achieved through several drafts. In designing the first draft, a qualitative preliminary evaluation of the most interesting questions has been performed. After this we have designed the questionnaire in detail according to the principles of Marbach and the psychological criteria of Converse and Presser [8] [13]. The final version of the questionnaire was achieved after several internal and external reviews.

The questionnaires have been filled in through personal and phone interviews. We interviewed managers for about 45 minutes. Participants were guaranteed anonymity and the information reported has been reviewed so that no individual person or company can be identified.

#### 3.2 The sample

The survey involved companies located in different countries: Italy, Switzerland, Canada and U.S. They operate in several business areas: consulting, services, software development, etc.

The average age of the interviewees is forty, nearly all (82%) are males and more than half (52%) are manager or R&D directors.

Concerning the level of knowledge and experience, most of them (60%) have a university degree (over Bachelor) and worked in their firm for less than five years (88%).

As showed in Figure 1, 40% of the companies have less than fifty employees and co-workers while 36% have more than a hundred. Moreover, 48% of them have been founded in the nineties.

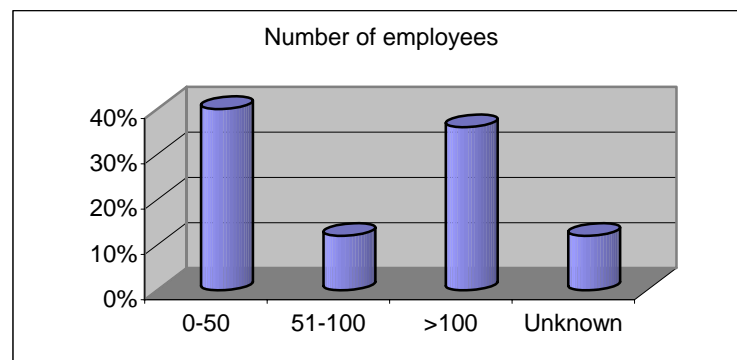
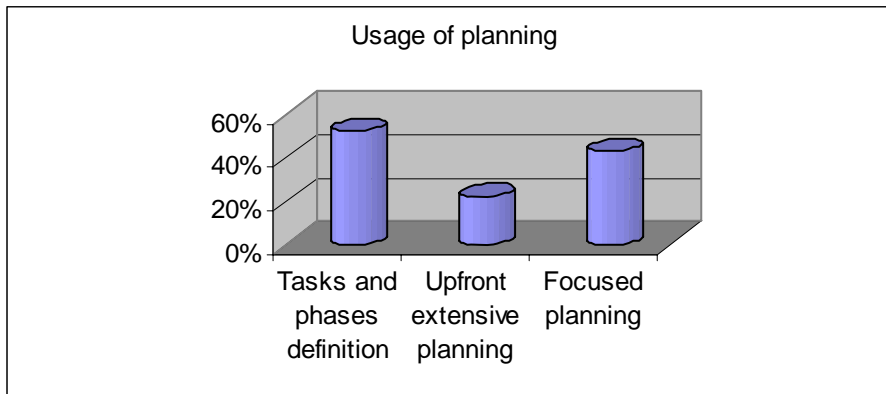


Figure 1: Number of employees

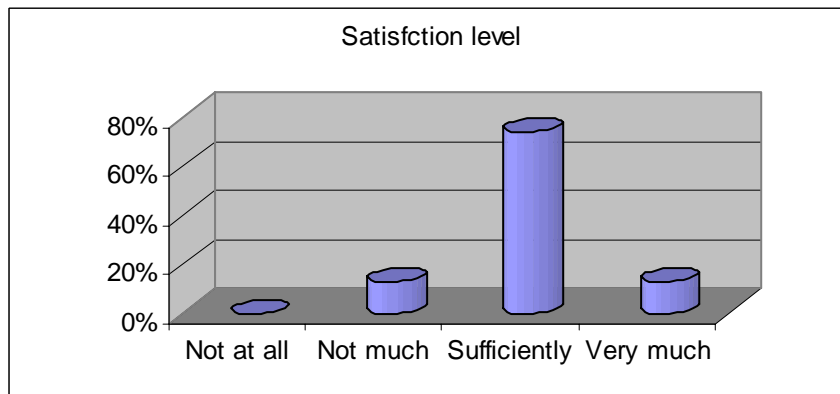
### 4. Analysis of the results

Most of the interviewed managers (85%) use planning tools in their projects. These tools are not the ones that are included in the AMs but they are the ones used in Plan-Based approaches and have been adapted to fit the agile environment. Managers use these planning tools at different levels (Figure 2): most of them (52%) plan the development process defining phases and/or tasks and keeping track of their evolutions; 22% perform an extensive use of upfront planning defining in details all the activities, the timeline, and how they have to be executed; finally, 43% do not use upfront planning for the entire project but plan only a limited set of critical functionalities and/or subsystems. In all these cases, the main motivations for the usage of such planning tools are: making the status of the development process available to middle and top managers, improve their ability to make estimates, method extensively used in the company.



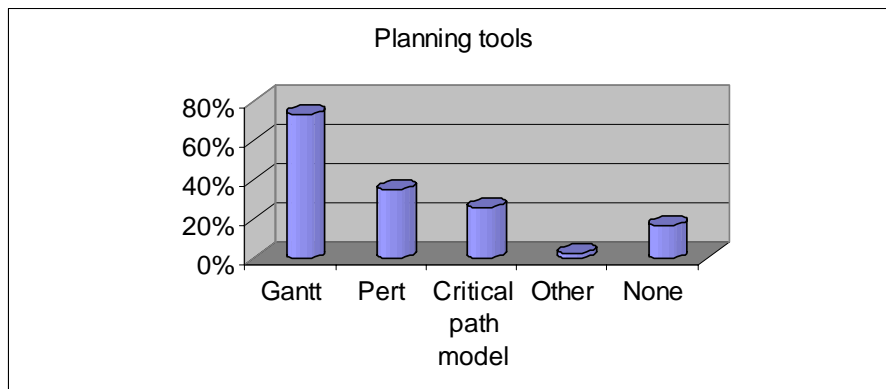
**Figure 2: Usage of planning**

According to Figure 3, more than 87% of the interviewed people are satisfied with their current planning techniques, however, nearly all (80%) would like to improve them. The main area of improvement is the integration of the planning techniques with the AM used. This could mean that the selection of the planning technique is often not linked with the development method used but with the techniques used in the company. Therefore, a smooth integration is of paramount importance to provide useful information to managers and do not interfere with the development.



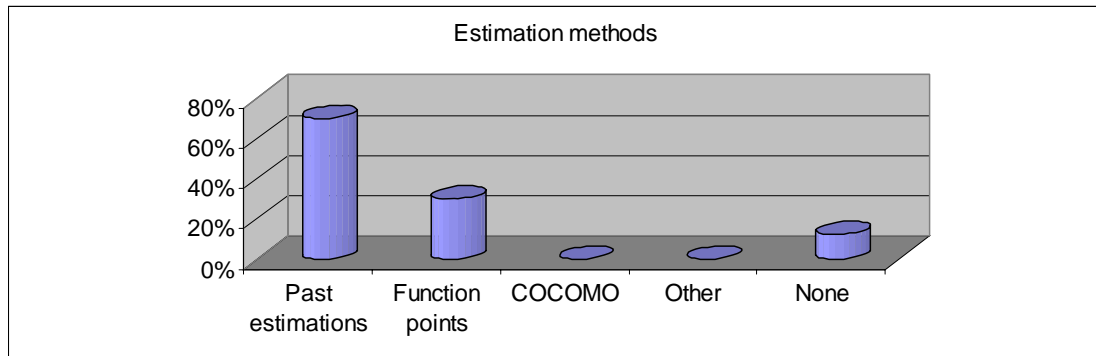
**Figure 3: Satisfaction level with the current planning**

The most used planning tools are: Gantt charts (73%), Pert charts (35%), Critical path model (26%). Only 3% of the managers use different tools and 17% do not use them (Figure 4).



**Figure 4: Planning tools used**

About the estimation methods, the most used ones are based on past estimations (70%) and function points (30%), while only 12% of them are not using any method. It is interesting that the COCOMO model and any other methods are not used at all (Figure 5).



**Figure 5: Estimation methods used**

The estimation methods used are linked with the methods that are widely used inside their companies. These are the methods that are used by middle and top managers to evaluate projects, make estimations and strategic decisions. Therefore project managers are forced to adopt such estimation approaches.

## 5. Conclusions

The survey analyzed the role of Plan-Based management approaches in Agile companies. According to the results, such companies widely adopt planning and estimation methods that are not Agile. This happens because AMs do not provide enough tools and middle and top managers are used to evaluate projects and make decisions on the base of more traditional approaches.

Unexpectedly, managers seem not to be free to choose the methods they consider appropriate for the specific project but the methods that are most commonly used in the company and the methods that the managers prefer. However, nearly all managers would like to improve their planning ability improving the integration of the tools with the AMs used.

This is only a preliminary survey. In order to generalize the conclusions of this paper, the results should be strengthened with additional data.

## 6. References

1. Abrahamsson, P., O. Salo, J. Ronkainen, J. Warsta, *Agile software development methods, review and analysis*, VTT Publications, 2002.
2. Anderson, D., *Agile Management For Software Engineering: Applying the Theory of Constraints for Business Results*, Prentice Hall, 2003.
3. Al-Rawas A., S. Easterbrook, "Communication problems in requirements engineering: A field study", *1<sup>st</sup> Westminster Conference on Professional Awareness in Software Engineering*, London, UK, 1 - 2 February, 1996.
4. Basili, V.R., *Software modeling and measurement: The Goal/Question/Metric paradigm*, Technical Report CS-TR-2956, Department of Computer Science, University of Maryland, College Park, MD 20742, 1992.
5. Beck, K., M. Beedle, A. Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. Martin, S. Mellor, K. Schwaber, J. Sutherland, D. Thomas, *Manifesto for Agile Software Development*, 2001, available online at: <http://www.agilemanifesto.org/>
6. Boehm, B., R. Tuner, "Using risk to balance agile and plan driven methods", *IEEE Computer*, 2003.
7. Chau, T., F. Maurer, G. Melnik, "Knowledge Sharing : Agile Methods vs. Tayloristic Methods", *In Proceedings of WETICE*, 302-307, Austria, 2003.
8. Converse, J.M., S. Presser, *Survey Questions: Handcrafting the Standardized Questionnaire*, Sage Publications, 1986.
9. Highsmith, J., *Agile software development ecosystem*, Addison Wesley, 2002.
10. Highsmith, J., What is Agile Software Development, *Crosstalk*, 2002.
11. Highsmith, J., A. Cockburn, "Agile software development: The business of innovation", *IEEE Computer*, September 2001.
12. Highsmith, J., A. Cockburn, "Agile software development: The people factor", *IEEE Computer*, November 2001.
13. Marbach, G., *Le ricerche di mercato*, Utet, 1996.
14. McConnell, S.C., *Rapid Development*, Microsoft Press, 1996.
15. Miller, R., *Managing Software or Growth without fear, control, and the manufacturing mindset*, Addison-Wesley, 2003.
16. Paetsch, F., A. Eberlein, F. Maurer, "Requirements engineering and agile software development", *Eighth International Workshop on Enterprise Security*, Linz, Austria, 9 - 11 June, 2003.
17. Poppendieck, M., T. Poppendieck, *Lean Software Development: An Agile Toolkit*, Addison-Wesley, 2003.

18. Siegel S., N.J. Castellan, *Nonparametric statistics*, McGraw-Hill, 1988.
19. Silverman, D., *Doing qualitative research*, Sage Publications, 2000.