

This is a reprint from a paper published in the Proceedings of the IADIS International Conferences
IADIS, <http://www.iadis.org>

MEASURING AGILITY AND ADOPTABILITY OF AGILE METHODS: A 4-DIMENSIONAL ANALYTICAL TOOL

Asif Qumer

*University of Technology, Sydney
PO Box 123, NSW 2007, Australia*

Brian Henderson-Sellers

*University of Technology, Sydney
PO Box 123, NSW 2007, Australia*

ABSTRACT

The software community is showing significant interest in agile software development methods and it is therefore desirable to have an analytical tool to evaluate current agile software development methods in practice. The purpose of this paper is to propose a 4-Dimensional Analytical Tool (4-DAT) for researchers and practitioners for the purpose of analysis and comparison of agile methods. 4-DAT will facilitate the examination of agile methods from four perspectives or dimensions: method scope characterization, agility characterization, agile values (agile manifesto) characterization and software process characterization. The tool is intended for use by software practitioners to compare and analyse agile methods. A report that is generated with the help of 4-DAT can be used for decision making regarding the adoption of an appropriate agile method.

KEYWORDS

Agility, Agile Methods, Software Process.

1. INTRODUCTION

The Agile Manifesto (2001) provides principles and values that characterize agile methods but there is no standard framework or tool for measuring agility and evaluating agile methods that can help in selecting a particular agile method for a particular project. The objective of this paper is to propose a 4-dimensional analytical tool (4-DAT) to compare and analyse such agile methods. Figure 1 shows the 4-dimensional visualization of a software development method.

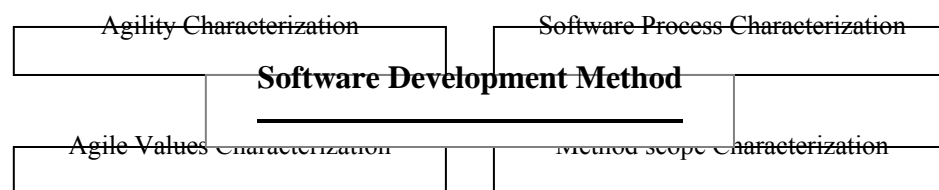


Figure 1. 4-Dimensional visualization

There are a number of analytical tools and frameworks (Kitchenham and Jones 1997, Williams *et al.* 2004, Boehm and Turner 2004a, Cuesta *et al.* 2002, Tran, Low and Williams 2004) that may be used to examine software development methods. The distinguishing feature of the proposed tool is that it specifically provides a mechanism to measure agility (degree of agility) of any method quantitatively at a specific level in a process and using specific practices. A report generated with the help of 4-DAT will assist organizations in making decisions about the selection or adoption of an agile method. The paper describes the 4-DAT in terms

of its four dimension. It also demonstrates the one of the key feature (agility measurement) of the tool with the help of an example. Finally it concludes with a discussion of our plans for future research.

2. METHOD SCOPE CHARACTERIZATION - DIMENSION 1

The method scope characterization dimension is a set of key scope items; having been derived by distilling a set of key scope items from selected agile methods: Extreme Programming (Beck 2000), Feature Driven Development (Palmer & Felsing 2002; Koch 2005), Adaptive Software Development (Highsmith 2000), Dynamic Software Development Method (Stapleton, 1997; Abrahamsson et al. 2002) and Scrum (Schwaber & Beedle 2002). This first dimension for the evaluation approach helps to compare the methods at a high level. Table 1 describes the first dimension of a 4-DAT.

Table 1. 4-DAT dimension 1

Scope	Description
1. Project Size	Does the method specify support for small, medium or large projects (business or other)?
2. Team Size	Does the method support for small or large teams (single or multiple teams)?
3. Development Style	Which development style (iterative, rapid) does the method cover?
4. Code Style	Does the method specify code style (simple or complex)?
5. Technology Environment	Which technology environment (tools, compilers) does the method specify?
6. Physical Environment	Which physical environment (co-located or distributed) does the method specify?
7. Business Culture	What type of business culture (collaborative, cooperative or non-collaborative) does the method specify?
8. Abstraction Mechanism	Does the method specify abstraction mechanism (object-oriented, agent-oriented)?

3. AGILITY CHARACTERIZATION - DIMENSION 2

Agility characterization is a second dimension of 4-DAT; and is a set of agility features (that are derived from the proposed agility definition, given in Section 3.1) with an agility measurement approach (see Section 3.2). Dimension two checks the existence of agility in agile methods at both a process level and a method practices level. Table 2 describes this second dimension of a 4-DAT. This is the only one of the four proposed dimensions of agility that is quantitative.

Table 2. 4-DAT dimension 2

Features	Description
1. Flexibility	Does the method accommodate expected or unexpected changes?
2. Speed	Does the method produce results quickly?
3. Leanness	Does the method follow shortest time span, use economical, simple and quality instruments for production?
4. Learning	Does the method apply updated prior knowledge and experience to learn?
5. Responsiveness	Does the method exhibit sensitiveness?

3.1 Notion of Agility - Features

The notion of agility is not new; but there is no rigorous or complete definition of agility. According to Dove (1997), agility is a very seductive word, evoking confusion with immediate and personal definitions for almost everyone. Wong and Whitman (1999) argue that agility refers to the effective response to rapid and unexpected change with flexibility, implying adaptability and versatility in the domain to respond to such unexpected changes. Conboy and Fitzgerald (2004, p.40) defines agility as: “the continual readiness of an entity to rapidly or inherently, proactively or reactively, embrace change, through high quality, simplistic, economical components and relationship with its environment”. According to Henderson-Sellers and Serour (2005), an agile entity improves over the period of time as it gains in experience and knowledge from its internal and external environment. Boehm and Turner (2004b) assert that “agility applies memory and history

to adjust to new environments, react and adapt, take advantage of unexpected opportunities, and update the experience base for future". Here we propose an independent definition of agility that defines the concept of agility in terms of flexibility, speed, leanness, learning and responsiveness; and covers the inadequacy of existing definitions.

Agility is a persistent behaviour or ability of a sensitive entity that exhibits flexibility to accommodate expected or unexpected changes rapidly, follows the shortest time span, uses economical, simple and quality instruments in a dynamic environment and applies updated prior knowledge and experience to learn from the internal and external environment.

3.2 Measuring Agility in Methods

Agility can be measured in terms of the five variables (features) described above: Flexibility (FY), Speed (SD), Leanness (LS), learning (LG) and responsiveness (RS) and that may exist in an object at some specific level or lifecycle phase or as a result of the practices used. The possible value of a feature or variable at a given time may be 0 or 1. For example, a software development method may encompass agility in the design phase, planning phase or in the requirements engineering phase – but not necessarily all three. We characterize the degree of agility (DA) for each of these phases as the fraction of the five agility variables that are encompassed and supported.

$$DA(\text{Object}) = (1/m) \sum m DA(\text{Object}, \text{Phase or Practices})$$

We use this approach to measure the degree of agility (at both phase and practice level) in one of the agile methods: Extreme Programming (XP). If any phase or practice of XP supports a particular agility feature, then we will allocate 1 point in that particular cell otherwise 0; and so on. Tables 3 and 4 demonstrate the agility measurement approach. First the agility is calculated in terms of the five features of agility for each phase of a process and then for each practice of a process.

Table 3. Agility values for XP based on (i) phases and (ii) practices

XP	Agility Features					Total
	FY	SD	LS	LG	RS	
(i) Phases						
Exploration	1	1	0	1	1	4
Planning	1	1	0	1	1	4
Iteration to release	1	1	0	1	1	4
Productionizing	1	1	1	1	1	5
Maintenance	1	0	0	1	1	3
Death	0	1	0	0	0	1
Total	5	5	1	5	5	21
Degree of Agility	5/6	5/6	1/6	5/6	5/6	21/(6*5)
(ii) Practices						
The Planning Game	1	1	0	1	1	4
Short Release	1	1		1	1	4
Metaphor	0	1	1	0	0	2
Simple Design	1	1	1	1	1	5
Testing	1	1	0	1	1	4
Refactoring	1	1	1	1	1	5
Pair Programming	1	0	0	1	1	3
Collective Ownership	1	0	0	1	1	3
Continuous Integration	1	1	1	1	1	5
40-Hours Week	0	0	0	1	0	1
On-site Customer	1	0	0	1	1	3
Coding Standards	1	1	1	1	1	5
Total	10	8	5	11	10	44
Degree of Agility	10/12	8/12	5/12	11/12	10/12	44/(12*5)

The final value of agility (degree) is shown in table 4 both for method's phases and practices, separately. These two assessments of the value of DA permit a separate ranking for both a process-based viewpoint and a practice-based viewpoint. A similar graph representing the agility at a phase level and practice level can be generated as well. There is, of course, no easy mathematical way of combining these two numbers and rankings and thus the decision-maker needs to include their own, often subjective, weightings to any evaluation of the most appropriate agile method to be adopted by their organization or for a particular project. This quantitative assessment is also supplemented by the other three, more qualitative, dimensions. We will give detailed assessment study of the full range of available agile methods in our next paper.

Table 4. Degree of agility of XP based on (i) phases and (ii) practices

Method	Degree of Agility (Phases)	Degree of Agility (Practices)
XP	DA based on phases - 0.70 = 21/30	DA based on practices - 0.73 =44 /60
*Note: Degree of agility of any object, method, phase, and practice ranges between 0 and 1.		

4. AGILE VALUES CHARACTERIZATION – DIMENSION 3

Agile values characterization, dimension three, is a set of six agile values; four of them are provided by the Agile Manifesto (2001); and the fifth agile value is provided by Koch (2005). The sixth value “keeping the process cost effective” is proposed here based on the study of several agile methods. Dimension three examines the support of agile values in different practices of agile methods. Table 5 describes this third dimension of a 4-DAT.

Table 5. 4-DAT dimension 3

Agile values	Description
1. Individuals and interactions over processes and tools	Which practices value people and interaction over processes and tools?
2. Working software over comprehensive documentation	Which practices value working software over comprehensive documentation?
3. Customer collaboration over contract negotiation	Which practices value customer collaboration over contract negotiation?
4. Responding to change over following a plan	Which practices value responding to change over following a plan?
5. Keeping the process agile	Which practices helps in keeping the process agile?
6. Keeping the process cost effective	Which practices helps in keeping the process cost effective?

5. SOFTWARE PROCESS CHARACTERIZATION - DIMENSION 4

Dimension four, is a set of four components of software process. There are two main components of software process: product engineering process and process management process. A product engineering process has further three categories: development process, project management process and support processes (Jalote 1997). Dimension four examines the practices that support these four processes in agile methods. Table 6 describes the fourth dimension of 4-DAT.

Table 6. 4-DAT dimension 4

Process	Description
1. Development Process	Which practices cover the main life cycle process and testing (Quality Assurance)?
2. Project Management Process	Which practices cover the overall management of the project?
3. Software Configuration Control Process / Support Process	Which practices cover the process that enables configuration management?
4. Process Management Process	Which practices cover the process that is required to manage the process itself?

6. CONCLUSION

In this paper, we have proposed a 4-DAT: a four dimensional tool for the comparative analysis of agile methods - or indeed any methods. The first dimension of the tool, method scope characterization, looks at the method from a perspective of its support for a specific software development. The second dimension, agility characterization, measures the existence (degree) of agility in a process of method at some phase and practice level. The third dimension, agile values characterization, checks the practices of a method for the support of agile values. Finally, the fourth dimension, software process characterization, analyses the method from the perspective of its support for four software process components. In future, we intend to refine and update the proposed 4-DAT and apply it to a highly detailed assessment study of the full range of available agile methods.

REFERENCES

- Abrahamsson, P., O. Salo, et al, 2002. *Agile software development methods. Review and analysis*. 107, VTT Publications 478, Finland.
- Abrahamsson, P., J. Wasta, et al, 2003. New Directions on Agile Methods: A Comparative Analysis. *25th International Conference on Software Engineering*. Portland, Oregon, pp.244-254.
- AgileManifesto. Manifesto for Agile Software Development.<http://www.agilemanifesto.org/>. Accessed 14 March 2005.
- Beck, K., 2000. *Extreme Programming Explained*. Boston, Addison-Wesley Pearson Education.
- Boehm, B. and R. Turner., 2004a. Balancing Agility and Discipline: Evaluating and Integrating Agile and Plan-Driven Methods. *Proceedings of the 26th International Conference on Software Engineering*. Washington, DC, USA, pp.718 – 719.
- Boehm, B. and R. Turner., 2004b. *Balancing Agility and Discipline: A Guide for the Perplexed*. Boston, Pearson Education, Inc.
- Conboy, K. and B. Fitzgerald., 2004. Toward a conceptual framework of agile methods: A study of agility in different disciplines. *Proceedings of the 2004 ACM Workshop on Interdisciplinary Software Engineering Research*. Newport Beach, CA, USA, pp.718 – 719.
- Cuesta, P., A. Gómez, et al, 2002. A Framework for Evaluation of Agent Oriented Methodologies. *Workshop on Agent-Oriented Information Systems (AOIS-2002)*.
- Dove, R., 1997. The Meaning of Life and The Meaning of Agility, Paradigm Shift International. <http://www.parshift.com/library.htm/>. Accessed 6 April 2005.
- DSDM. DSDM Consortium, Dynamic Systems Development Method Ltd. <http://www.dsdm.org/>. 20 May 2005.
- Henderson-Sellers, B. and Serour, M.K., 2005. Creating a dual agility method - the value of method engineering. *J. Database Management*, Vol. 16, No. 4, pp.1-24.
- Highsmith, J. A. I., 2000. *Adaptive Software Development: A Collaborative Approach To Managing Complex Systems*. New York, Dorset House Publishing.
- Jalote, P., 1997. *An Integrated Approach to Software Engineering*. New York, Springer-Verlag.
- Kitchenham, B. A. and L. Jones., 1997. Evaluating Software Engineering Methods and Tool part 5: the Influence of Human Factors. *ACM SIGSOFT Software Engineering Notes*, Vol. 22, No. 1, pp.13-15.
- Koch, A. S., 2005. *Agile Software Development: Evaluating the Methods for Your Organization*. London, Artech House, Inc.
- Palmer, S. R. and J. M. Felsing., 2002. *A Practical Guide to Feature-Driven Development*. Upper Saddle River, Prentice-Hall Inc.
- Schwaber, K. and M. Beedle., 2002. *Agile Software Development with SCRUM*. Upper Saddle River, Prentice-Hall Inc
- Stapleton, J., 1997. *DSDM: The Method in Practice*. Reading, MA, Addison-Wesley
- Tran, Q.-N. N., G. Low, et al, 2004. A Preliminary Comparative Feature Analysis of Multi-agent Systems Development Methodologies. *6th International Bi-Conference Workshop on Agent-Oriented Information Systems*. New York, USA, pp.386-398.
- Williams, L., W. Krebs, et al, 2004. Toward a Framework for Evaluating Extreme Programming. *Empirical Assessment in Software Engineering*. Edinburgh, pp.11-20.
- Wolak, R. G., 2001. System Development: Research Paper 1, SDLC on a Diet. <http://www.itstudyguide.com/papers/rwDISS725researchpaper1.htm/>. Accessed 25 April 2005.
- Wong, S.-P. and L. Whitman., 1999. Attaining Agility At The Enterprise Level. *Proceedings of the 4th Annual International Conference on Industrial Engineering Theory, Applications and Practice*. San Antonio, Texas, USA.