

WHAT AGILE TEAMS THINK OF AGILE PRINCIPLES



JYOTIRMOY PATGIRI

IN THE MID-1990

- ❑ The prescribed means of keeping software development projects out of trouble and on schedule was to follow a heavyweight software development methodology consisting of a complete requirements document, including architecture and design, followed by coding and testing based on a thorough test plan.
- ❑ The philosophy was often summarized as “Do it right the first time.” Common belief among software engineers at the time was that projects run into trouble when they do not strictly adhere to a methodology, and, if only they did, all would be well in reality, all was rarely well.
- ❑ To ease up the process, lightweight software development methodology was introduced. First was **EXTERME PROGRAMING** in 1999s.
- ❑ This was followed by Adaptive Software Development(ASD), Crysta, Dynamic Systems Development Method(DSDM), Feature Driven Development(FDD)and Scrum.

IN THE EARLY-2000

- ❑ In February 2001, rather than focus on their differences and the “competitive advantage” of their own methodologies, 17 creators and supporters of the lightweight methodologies gathered in Snowbird, UT, to discuss their common interests and philosophies, coining the term “agile software development” to describe their methodologies.
- ❑ They created the **manifesto** (public declaration of policy and aims) focusing on
 - ✓ *Individuals and interactions*
 - ✓ *Working software*
 - ✓ *Customer collaboration*
 - ✓ *Responding to change*
- ❑ The Agile Manifesto and the agile principles thus began to serve as a rallying cry for some and the bull’s-eye in the dartboard for others.
- ❑ So how well do the Agile Manifesto and its 12 principles have matured and evolved?
How do agile teams regard the principles today?

SURVEY METHODOLOGY

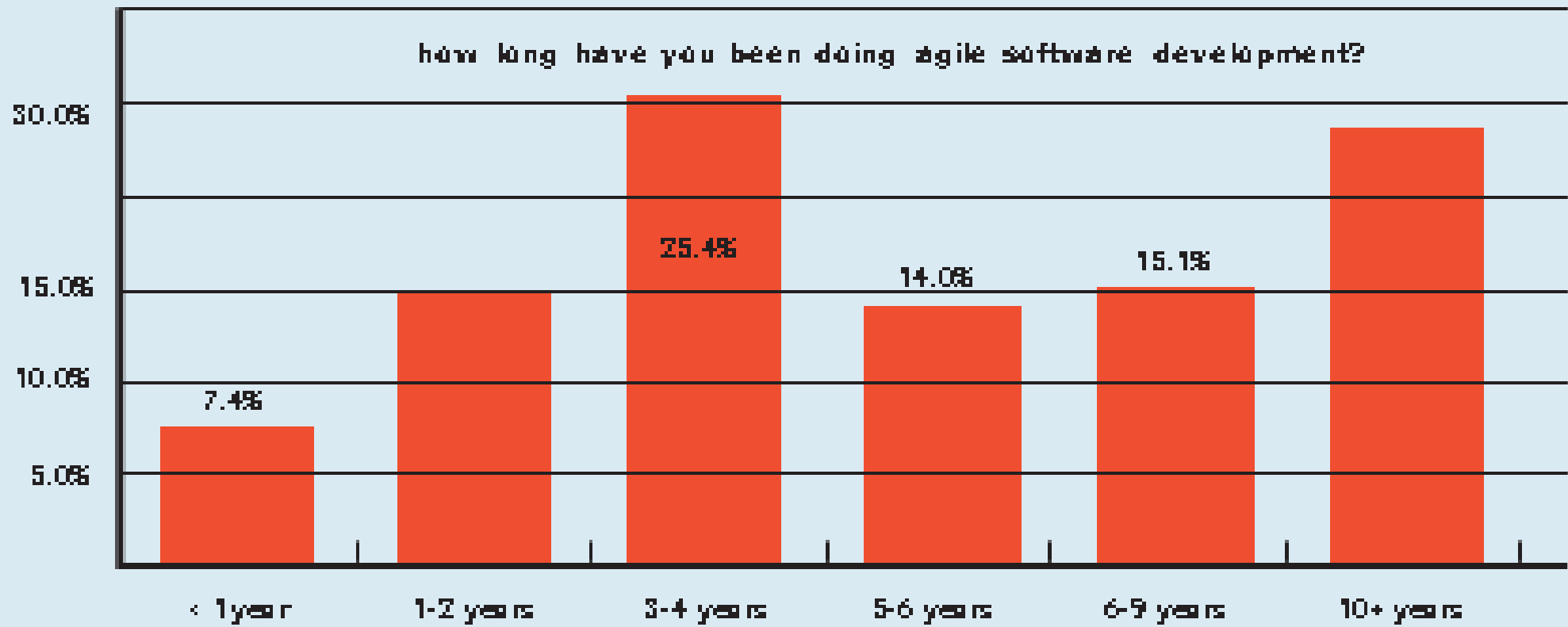
- ❑ Two surveys in 2010 at North Carolina State University to weigh the community's view of the principles and use of associated practices.
- ❑ Administered them through surveymonkey.com, advertising the first survey on a number of agile-related user groups (such as those on Yahoo! and LinkedIn).
- ❑ Additionally, emailed approximately 100 personal contacts, inviting them to participate and forward the survey to their colleagues.

“How important is this principle that comes from the original agile principles authored in 2001 for agile teams in 2010?” (1=not very important; 5=essential, the team is not agile if it doesn't follow this principle)

- ❑ Respondents were primarily from North America (59%) and Europe (29%).

SURVEY OUTCOME

Survey respondents' experience with agile software development.



TIER ONE (AVERAGE SCORE : MEAN 4.6)

□ Principle 1

“Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.”

□ Principle 3

“Deliver working software frequently, from a couple of weeks to a couple of months, with a preference for the shorter timescale.”

Comment

□ Respondents' commentary emphasized delivery of a solution with “high business value” to a customer early and often, along with willingness to respond to feedback.

□ Few respondent suggested that **principles 1 and 3 were probably redundant**, a view supported by statistics based on overall survey responses.

TIER TWO (AVERAGE SCORE : MEAN 4.5)

□ Principle 5

“Build projects around motivated individuals. Give them an environment and support they need, and trust them to get the job done.”

Comment : emphasized the need to empower and respect motivated individuals while making them “feel they can make a difference and [are] part of building something out of the ordinary.” Some respondents said providing the “support they need” included removing obstacles so the team could operate efficiently.

□ Principle 7

“ Working software is the primary measure of progress.”

Comment : This principle has got the most comments. Most of the teams give priority to quickly reach “get done” criteria. It is important to focus on quality and thorough testing before declaring a project as done.

□ Principle 12

“The team regularly reflects on how to be more effective, tuning and adjusting its behaviour accordingly.”

Comment : Strong Agreement. for feedback and creating a culture of continuous improvement and building respect.”

TIER THREE (AVERAGE SCORE : MEAN 4.3)

□ Principle 9

“Continuous attention to technical excellence and good design enhances agility.”

Comment : Respondents gave strong support for this principle but provided no further commentary or clarification of their views.

TIER FOUR (AVERAGE SCORE: MEAN 4.2)

□ Principle 2

“Welcome changing requirements even late in development; agile processes harness change for the customer’s competitive advantage.”

Comment :Some commenters on principle 2 suggested a project’s requirements should change only at the beginning of each iteration. Some project may not complete due to constantly changing priorities.

□ Principle 10

“Simplicity, the art of maximizing the amount of work not done, is essential.”

Commenter described principle 10 as “Build great software...that addresses users’ needs without unnecessary features.

TIER FIVE (AVERAGE SCORE : MEAN 4.1)

❑ Principle 4

“Business people and developers must work together daily throughout the project.”

Comment :

- ✓ Developers (often seen as those writing the code) should not be the only ones to work with businesspeople (a.k.a product owners). Rather, the whole team, including user interface analysts, testers, project managers, developers, and businesspeople should collaborate.
- ✓ Others commenters said, “Every day often isn’t realistic, nor is it necessarily needed.”

❑ Principle 6

“The most effective method of conveying information to and within a development team is face-to-face conversation.”

Comment : Principle 6 was generally supported, though some commenters said the “requirement for face-to-face conversation is a severely limiting factor for distributed teams, and it seems to be a generational issue as well.” In today’s connected world, synchronous communication through instant messaging, Voice over IP, and WebEx may effectively stand in for face-to-face communication.

❑ Principle 8

“Agile processes promote sustainable development; sponsors, developers, and users should be able to maintain a constant pace indefinitely.”

Several representative comments on principle 8 indicating the commenters’ negative experience

- ✓ “Agile does not promote sustainable development but increases the kind of focus that leads to burnout”
- ✓ “Sustainable pace is extremely important, but we also sometimes have to slow down and think about things a little”
- ✓ “The team should have dedicated exploratory study time that contributes to its ability to produce innovation”

Tier Six (Average Score : mean 3.8)

❑ Principle 11

“ The best architectures, requirements, and designs emerge from self organizing teams.”

Comment : “You really need to do some systems engineering when building large systems,” should understand how the product “contribute[s] to the larger goals of the [user] organization and must have a release plan(proper documentation)

CONCLUSION

- ❑ The recent introduction of the lean software development **Kanban** practice removed the notion of iterations for many teams. With kanban, a feature can begin at any time if the “pull system” indicates the team has the capacity to start new work. As a result, Kanban teams often lack defined iterations.
- ❑ An agile practice that survey respondents said was left off the list was the **“spike”**; that is, teams do spikes, when they do not know enough about a feature to effectively estimate the resources needed for its implementation. A spike is a timeboxed experiment that allows developers to learn just enough about something unknown about a feature implementation (such as a new technology) to be able to estimate the effort required to deliver the feature.
- ❑ Survey respondents (Surprisingly even distributed teams) generally emphasized that nothing beats face-to-face for verbal and non-verbal communication alike, and wanted principle 6 to represent the Ideal practice.
- ❑ Overall, the results of both surveys suggested overwhelming support for the original principles, even after more than 10 years of use.

The background is a dark green gradient. In the four corners, there are decorative white line-art patterns resembling circuit traces or neural network connections, with small circles at the end of the lines.

```
PRINT ("THANK YOU !")
```

```
PUTS "THANK YOU"
```

```
< ? PHP ECHO '<P>THANK YOU !</P>; ? >
```

```
SYSTEM.OUT.PRINTLN( "THANK YOU !");
```