A common misconception in software project management is that in order to properly follow the practices outlined in *A Guide to the Project Management Body of Knowledge (PMBOK® Guide)---Third Edition* (Project Management Institute [PMI], 2004), we must use a nonagile, or waterfall, approach. This paper attempts to correct that fallacy and show that agile projects still follow the project life cycle and processes as outlined by the *PMBOK® Guide*. First we must examine the origins of the *PMBOK® Guide*, the book most often used as a reference by project managers. Then we’ll see how its project life cycle and processes correlate to those in an agile project.

The Project Management Institute and the *PMBOK® Guide*

The Project Management Institute was founded in 1969 at the Georgia Institute of Technology by five volunteers: James Snyder, Gordon Davis, Eric Jenett, A.E. Engman, and Susan C. Gallagher (PMI, 2005). Its original purpose was to form an organization in which members could share their experiences in project management and discuss issues. The purpose has expanded today to advancing practice knowledge and application in the profession of project management.

To this end, in 1983 the PMI created a publication entitled “PMI Special Report on Ethics, Standards, and Accreditation.” The “Standards” portion of this document was the “Project Management Body of Knowledge.” In 1996 the first edition of the *PMBOK® Guide*, a book that outlined project management knowledge areas, processes, and practices, was published. The *PMBOK® Guide* became a standard for generally recognized good practices in project management. Now in its third edition, the *PMBOK® Guide* has sold more than a million copies worldwide. For years this has been the de facto archetype that all project managers follow—not just software project managers.

Although the *PMBOK® Guide* does not dictate methodology, many software project managers nevertheless began to associate the waterfall model with the processes outlined in the *PMBOK® Guide*. Perhaps it was because waterfall was the prevalent methodology at the time, or perhaps it was because the waterfall model provided a framework that supported all of the *PMBOK® Guide* practices. Whatever the reason, it has been a hard misconception to shake, even though the third edition of the *PMBOK® Guide* makes it very clear that it is up to the reader to determine what processes are most appropriate to use in their situation.

Indeed, the *PMBOK® Guide* has paradoxically become broader in its context even as it becomes more detailed in its processes and practices. As an important and noted change from the 2000 edition, the third edition states clearly that “there is no single best way to define an ideal project life cycle” (PMI, 2004, p. 20). It goes on to say that “the project manager, in collaboration with the project team, is always responsible for determining what processes are appropriate, and the appropriate degree of rigor for each process, for any given project” (PMI, 2004, p. 37). Although the 2000 edition may have made it difficult to make a case showing how agile practices are in keeping with best practices as outlined in the *PMBOK® Guide*, the third edition makes it easy.

It is not only the *PMBOK® Guide* that is clear in its support of the validity of the newer agile methodologies. PMI’s magazine *PM Network®* started talking specifically about agile practices in April 2005, when Peter Fretty’s feature article “Reconciling Differences” shone a spotlight on how agile practices had improved productivity, quality, and customer satisfaction at Shine Technologies (Fretty, 2005). That was the first of several articles that have since appeared in the magazine touting agility. PMI is also supporting the training of its project managers in Agile Project Management courses and presentations in PMI-sponsored programs such as PMI Seminars World®, PMI Global Congresses®, and chapter symposiums and conferences.
PMI does not advocate any particular methodology. It only supplies a standard of good project management practices, and whether individuals choose to follow a waterfall or an agile approach, the PMBOK® Guide will support them both. You don’t have to cast aside your PMP designation to be agile. All you need to do is change how you implement the practices (and sometimes you may need to rethink where you’ve been placing your focus).

Project Life Cycle

The PMBOK® Guide defines the project life cycle as “a collection of generally sequential project phases whose name and number are determined by the control needs of the organization or organizations involved in the project” (PMI, 2004, p. 368). These phases are a collection of logical groupings of related activities that usually culminate in a deliverable. The PMBOK® Guide describes their sequence as beginning with an initial phase, followed by a series of intermediate phases, and ending in a final phase (Exhibit 1). Processes that aid in the completion of the deliverable are performed in each phase.

Exhibit 1--Project Life Cycle Phases (PMI, 2004, p. 23)

In traditional approaches to software development, these phases have typically followed the waterfall methodology (Exhibit 2). For example, one phase might be Design, and the deliverable for this phase would be some type of a system design document. Many times sign-offs are required before proceeding to the next phase, but this is the result of the methodology and not a mandate by the PMBOK® Guide.

Let’s look at how this translates to an agile project life cycle. In the definition “a collection of generally sequential project phases,” the word “sequential” is often the biggest stumbling block. This is due to the popular misconception that “sequential” equals “waterfall.” The agile project life cycle has sequential project phases that we call iterations, with the deliverable in each iteration being working code. However, all of the processes that are typically done sequentially—analysis, design, code, test—are done within a single phase in agile projects to produce an increment of code.

You may even want to consider these iterations as subphases of a larger phase known as the release, where the major deliverable is set of those increments of code such that they can be put into production and/or delivered to the customer.

The second part of the project life cycle definition points out that the number (and in our agile interpretation, the duration) of the phases is “determined by the control needs of the organization.” In agile projects, the number of iterations is decided on by the customer, based on what they define as the minimum amount of functionality deemed acceptable for a release. The length of the iterations—generally between 1 and 4 weeks—is determined by the “control needs” of the customer (or customer representative)—that is, how often the customer will want to change what the delivery team is working on. The volatility of the industry, the amount of risk, and the clarity of the vision are all factors that affect the length of the iteration and define the organization’s need for flexibility.

In the initial phase of agile projects, there is a planning process that is part of the first iteration, and the deliverable is the high-level plan for the project (and in most cases, the first increment of working code is delivered as well). Intermediate phases in agile projects are the releases and/or iterations where additional features are delivered in the form of working code. The final phase in an agile project is the hardening or production-readiness phase, where process activities that prepare the product for delivery are conducted, along with the final project retrospective and other closing processes.

The PMBOK® Guide states that the transition from one phase to another usually involves some type of technical transfer or hand-off (PMI, 2004, p. 20). This is the type of phrasing that might lead readers to believe that only a waterfall methodology is appropriate when following PMBOK® Guide practices. However if we interpret “hand-off” to mean that there is a hand-off of an increment of code to the customer to use as they see fit, then agile is still in keeping with the basic tenets of the PMBOK® Guide. At the end of each iteration, an increment of code is completed by the team and reviewed by the customer. Regardless of the outcome, the next iteration begins as planned (unless the project is cancelled)—only the content of the work for that iteration is subject to change.

Because of this regular rhythm of incremental delivery, many have proffered that each iteration of an agile project is itself a project, having a start and stop date, and delivering a product as a result. I disagree with this assessment, however, believing it to be colored by years of waterfall practice. The waterfall methodology outlines the processes of analysis, design, coding, testing, and deployment, which were all done as part of a project. Because these are things that are all done within an iteration in agile, the logical assumption for many was that an iteration equaled a project. But an iteration is more properly referred to as a phase or subphase of the project, if using PMBOK® Guide terminology. Projects are “undertaken to create a lasting outcome” (PMI, 2004, p. 5), with the project team generally remaining the same until the project ends. In agile projects the delivery team is kept together from iteration to iteration, with each delivered increment an enhancement and/or evolution of the previous increment. The PMBOK® Guide refers to this as “progressive elaboration” (PMI, 2004, pp. 6, 8), and includes this as a characteristic of some projects. It defines agile projects perfectly.

The PMBOK® Guide notes that each phase (iteration) should have a formal initiation outlining what deliverables are expected in that phase, and a formal review at the end to conclude the phase with permissions obtained to continue or a decision made to stop the project. Agile iterations work on this premise as well. The initiation is done by the customer, whether informally with a verbal committal that work should begin or approval that work should continue, or formally through the use of contracts. Agile iterations begin with a planning meeting to define what will be completed in the iteration and end with a review to learn from the events and obtain customer acceptance of the features delivered. During the
review, the project can be cancelled or approved to continue, or a release can be requested and implemented immediately or implemented in the next iteration.

Exhibit 3 shows how the project life cycle as depicted in the PMBOK® Guide can be mapped to the agile project life cycle. In fact, the agile project life cycle is simply a fractal, as illustrated by the expanded phases in the figure. An agile project can be made up of multiple releases or periods of calendar time (quarters are commonly used), which in turn are made up of iterations in which teams create the increment of working code. Each has an initial phase in which planning is a key process, intermediate phases, and a final phase in which reviews and retrospectives are key processes.

Exhibit 3--Phases and Subphases in an Agile Project Life Cycle (Agile Fractal).

One area in which agile projects and the PMBOK® Guide disagree is in the involvement of the stakeholders. In agile projects, there is an expectation of the active involvement of a customer or customer representative throughout the duration of the project. This individual sets the direction for the product at the outset of the project, and refines that vision at the beginning of each iteration, with the same high level of influence each iteration over the characteristics of the product until the product is released. The PMBOK® Guide takes the view that stakeholder influence occurs up front and then declines throughout the rest of the project (PMI, 2004, p. 21). In agile projects, however, the stakeholders’ influence remains strong and does not decrease until the product is released and the project is over. Agile welcomes change and provides a framework that manages that change through the use of iterative and incremental development with regular customer feedback, reviews, and retrospective analysis.

Project Management Processes

Walter A. Shewhart’s Plan-Do-Check-Act (PDCA) model, also known as the “Shewhart cycle,” was made popular by the quality guru, Dr. W. Edwards Deming. The PMBOK® Guide acknowledges this iterative method of continuous improvement and mapped its project management process groups to the PDCA cycle (Exhibit 4). These project management process groups defined by the PMBOK® Guide are Initiating, Planning, Executing, Monitoring and Controlling, and Closing. The graphic on the right in Exhibit 4 is from the PMBOK® Guide (PMI, 2004, p. 40).
Exhibit 4--Plan-Do-Check-Act and the PMBOK® Guide Process Groups

The process groups are not phases, but rather they are an integrated set of processes applied iteratively throughout the project and revised as needed. Like the project life cycle, we can map the process groups to the agile fractal—at the overall project level, at the release level, and at the iteration level (see Exhibit 5).

<table>
<thead>
<tr>
<th>Agile Fractal</th>
<th>Initiation</th>
<th>Planning</th>
<th>Executing</th>
<th>Monitoring &amp; Controlling</th>
<th>Closing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project</td>
<td>Business case or feasibility study</td>
<td>Project kickoff and visioning meeting</td>
<td>Iterative and incremental delivery of working software</td>
<td>Regular reviews of deliverables, progress, and process</td>
<td>Project retrospective</td>
</tr>
<tr>
<td></td>
<td>Release roadmap planning</td>
<td>Release roadmap planning meeting</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Release</td>
<td>Roadmap and release definition</td>
<td>Release planning meeting</td>
<td>Iterative and incremental delivery of working software</td>
<td>Regular reviews of deliverables, progress, and process</td>
<td>Release retrospective</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Iteration</td>
<td>Iteration planning meeting</td>
<td>Iteration planning meeting</td>
<td>Work features through to completion (includes testing)</td>
<td>Task boards, burndown charts, daily stand-ups, acceptance of completed features</td>
<td>Iteration demo, review, and retrospective</td>
</tr>
</tbody>
</table>

Exhibit 5--Process Groups and the Agile Fractal

Conclusion

The PMBOK® Guide is a standard for generally recognized good practices in project management. Unfortunately, misconceptions still exist regarding the type of methodologies, as outlined in the PMBOK® Guide, that can be used to implement these practices. It is perfectly fine to use an agile approach—you can do so and still be in keeping with the recommendations in the PMBOK® Guide.
The *PMBOK*® Guide outlines project life cycle phases that correspond to agile releases and/or iterations. An agile project can be made up of multiple releases or calendar quarters, which in turn are made up of iterations. The initial phase in an agile project includes planning processes and usually a delivery of an increment of code; the final phase includes hardening and production readiness processes as well as a final project retrospective. All intermediate phases focus on the delivery of increments of working code. Agile projects use Shewhart’s Plan-Do-Check-Act cycle as part of the integrated processes in each agile fractal.

References


