Can Agile and Traditional Project Management be Partners?
Integrating the Agile Methods with Project Management Best Practices

By

Paul H. Lohnes, PMP **Ms. Cheryl A. Wilson, CCEP, PMP, PMI-RMP**

Executive Summary

With Agile/Scrum being the latest methodology "du jour" for the management of software and IT projects, this discussion centers on the question:

> *can this newest of project management environments be adapted to co-exist with the current industry methods of more traditional project management discipline practices?*

An initial review of the current literature on the topic of agile project management, its history, guiding values, and conceptual principles will begin this discovery process. With this background, the comparison with current traditional project management practices can take place in the correct context.

The strengths and weaknesses of both the Agile/Scrum methodologies in comparison to more traditional project management methods and process will be explored so that each discipline can be more accurately assessed by the reader in applying the correct approach to projects in their particular environment. Next, the areas of agreement and contention can be dissected appropriately with a concluding discussion of the possible points of resolution whereby Agile and traditional project management can peacefully co-exist.

The article will also list the next steps that an organization must take to understand if and how Agile methods and PMBOK-style project management can be integrated inside their unique organizational profiles.

Table of Contents

What is Agile?....2
Bits or Atoms? Reasons for Agile's Success....3
So What is Agile Project Management?....4
Strengths and Weaknesses of Agile....5
Strengths and Weaknesses of Project Management CBP....6
Comparison of Agile and Project Management CBP....7
Major Differences between Agile and Project Management CBP....8
Agreement and Points of Contention....9
Steps an Organization Should Take in Choosing a Development Philosophy....9
References....9
Authors' Biographies....10

The answer can be found initially in the Agile Manifesto of Feb. 13, 2001, well, let the Agile Alliance state this in their own words:

> *On February 11-13, 2001, at The Lodge at Snowbird ski resort in the Wasatch mountains of Utah, seventeen people met to talk, ski, relax, and try to find common ground, and of course, to eat. What emerged was the Agile Software Development Manifesto. Representatives from Extreme Programming, SCRUM, DSDM, Adaptive Software Development, Crystal, Feature-Driven Development, Pragmatic Programming, and others sympathetic to the need for an alternative to documentation driven, heavyweight software development processes convened.* [1]

This manifesto espouses the guiding values and principles of the Agile philosophy:

Agile is guided by 4 values:

1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan

and supplemented by 12 principles:

- Customer satisfaction by rapid delivery of useful software
- Welcome changing requirements, even late in development
- Working software is delivered frequently (weeks rather than months)
- Working software is the principal measure of progress
- Sustainable development, able to maintain a constant pace
- Close, daily cooperation between businesspeople and developers
- Face-to-face conversation is the best form of communication (co-location)
- Projects are built around motivated individuals, who should be trusted
- Continuous attention to technical excellence and good design
- Simplicity
- Self-organizing teams
- Regular adaptation to changing circumstances

By 2001, there were already many 'rapid application development' methodologies in existence [2]; what was lacking, it appeared, was a concise statement of value and focus. The Agile Manifesto provided this for the above listed RAD methodologies whose representative practitioners became signatories to the Manifesto.

The current literature on Agile methods

Kent Beck: creator of eXtreme Programming (XP) [3]
Jeff Sutherland, PhD: creator of Scrum [4]
Scott Ambler: creator of RUP, AUP, and EUP [5]
Alistair Cockburn: creator of Crystal [6]
Jim Highsmith: prime mover behind APM (Agile Project Management) [7]

have two consistent themes involving the use of Agile methodologies and the best way to ensure their benefits to the organizations using Agile methods:

1. Agile is best used in developing software for computers, and
2. The team size and co-location of the project/product owner and team are crucial

The frustration of the Agile signatories were evident in the manner in which the Agile Manifesto swung the philosophy pendulum in completely the other direction. If you read the guiding values and principles closely, they repudiate almost word for the manner in which IT project management and software product development had been attempted to the point of the manifesto signing.

No requirements gathering? No documentation? No contracts? No infinitely detailed deliverables scope? The Agile manifesto was a cannon shot across the bow of the current best practices practitioners reminiscent of the scream from the movie Network:

> "I am mad as hell, and I am not going to take it anymore…"

However, did the Agile signatories really think this through before embarking on such a radical change, or was it the radical nature of the manifesto the end in itself? Figuring an answer to that would take some time and hard evidence of purpose and intent, but the results seem to bear out the affirmative. The methodologies that streams from the point of agreement brought on by the Agile manifesto have yet to stop ringing in the halls of traditional project management.

The point from here on is:

> "can Agile peacefully co-exist with project management current best practices?"

This answer is so important for many project management practitioners and PM-contract firms since the choices made will determine the capabilities to achieve successes on the projects under assignment. If Agile can be applied to all forms of project management, what does this mean for the project management current best practices model? If Agile is applicable to only a small sub-section of project or product development solutions, what are the questions a project manager or firm needs to ask in order to determine if and where these seemingly orthogonal philosophies can be applied according to their success potentials?

Bits or Atoms: reasons for Agile's success?

Building on the frustrations now evident at the basis for the Agile Manifesto's creation, where did the project management current best practices seem to go "off-rails," when this previously venerated project management philosophy had worked so well in the heavy construction, civil engineering, and physical product development environments? The answer to this question lies in understanding one of the causes for such an alarmingly high rate of failure when IT projects are attempted using this previously universal form of project management philosophy.

The themes or indications of application for Agile methods illustrate a point that seems to have been missed by many project management thinkers and philosophers when understanding why so many IT projects fail. For proof of this statement, the reader is directed to the Standish Group's often quoted "project failure studies of 2003, and 2009. [8]. In addition, the UK Government has done significantly detailed studies of project failure. A simple Google of these search parameters will illustrate this point.

As mentioned previously, a strong basis for the project management 'current best practices' of stringently defining requirements with copious amounts of documentation had their roots in the construction industry of the 1930-1960's when humans were doing some of the most complex building projects outside the ancient projects of antiquity. Software development, however, is an entirely different endeavor from 'hard construction,' and to apply project management concepts involving the creation of physical artifacts to the construction of computer applications appears to be a fundamental misunderstanding of the core difference between these types of projects, and that is:

the physicality of the deliverables,

where "hard construction" delivers physical artifacts (atoms), and software projects deliver non-physical artifacts (bits).

The 17 signatories of the 2001 Agile Manifesto were software engineers attempting to deal with the frustration, overhead, strict, and probably inappropriate nature of normal project management methods when applied to the computer software development environment. The need to define requirements when producing 'atoms' overly constrained the software project where deliverables are 'bits.' Agile style methods, under conditions to be discussed later in this white paper, appear to solve the problems of managing work product when the deliverable has not physical or analog, but digital in nature.

This difference in deliverable physicality is at the heart of the current Agile revolution where software project management is finding its own voice in managing "bits" to a successful outcome requires very different methods than managing or producing "atoms".

With an understanding of Agile, its values and principles as a foundation, the next discovery phase needs to uncover the move of Agile concepts into the larger realm of general, or non-IT project management. This leads to the question,

So what is Agile Project Management (APM)?

Is APM an attempt to apply the values and principles of the Agile philosophy to the problems of "atoms-oriented" project management? Can these values and principles hold up when the deliverable is not "bits," but atoms in varying shapes and sizes? To put a finer point on it: could the Millau Viaduct have been built using the values, principles, and practices of Agile/Scrum or Agile/AUP? The reader is left to consider his/her answer to this question:

Are the values and principles of APM different from those of the Agile Manifesto?

After the initial release and impact of the Agile Manifesto in the software industry, several Manifesto originators and others from the project management discipline attempted to capture the Agile philosophy which they codified in the "Declaration of Independence for modern management," which can be found at the web site [9] of Dr. Alistair Cockburn, an Agile Manifesto original signatory:

"We ...

- increase return on investment by making continuous flow of value our focus.
- deliver reliable results by engaging customers in frequent interactions and shared ownership.
- expect uncertainty and manage for it through iterations, anticipation and adaptation.
- unleash creativity and innovation by recognizing that individuals are the ultimate source of value, and creating an environment where they can make a difference.
- boost performance through group accountability for results and shared responsibility for team effectiveness.
- improve effectiveness and reliability through situationally (sic) specific strategies, processes and practices."

A rather thorough search of the Internet has not indicated much more than this initial declaration of applying agile philosophies to general project management; however there are some recent books that appear to be approaching the point of such application:
Jim Highsmith, a Agile Manifesto original signatory, has the most recent book on Agile philosophies as they can be applied to general project/product development.

Robert Wysocki, a Declaration of Independence co-developer, has an updated project management book on how to use more agile-like processes within the 'current best practices' discipline.

John C. Goodpasture takes an integrated approach to making Agile philosophy apply across general project management boundaries.

Books by:
Ken Schwaber, Michele Sliger, Stacia Broderick, Mike Cohn, Roman Pichler, and on discuss the application of the Agile philosophy to software or IT projects.

What is common amongst most of the practitioners of the Agile philosophy of project management is that is works well in certain areas of application, but there have been few case studies of application when the deliverables are "atoms" and not "bits." The project management blogosphere has had significant discussions on the use of Agile-based methods for large projects such as the Boeing 787 Dreamliner, the new 3rd lane of the Panama Canal, and the previously mentioned, Millau Viaduct in France.

Research and discussions on applying Agile-style methodologies to large, more complex projects are continuing with some early results showing that regardless of the task, the most effective team size for an agile type project is 7 +/- 2. This is also known as the "2 pizza rule," [10] a phrase coined by Jeff Bezos, Amazon's Founder, when describing when a team size was too large.

The aspect of Agile's appropriateness to general project management endeavors still seems stalled at the physicality boundary. Agile has a recent history of applicability in small, very contained project teams where the primary tenets of the Agile philosophy as embodied in such methodologies as Scrum can play out in a supported environment. Scrum has immerged as the predominant implementation of the Agile philosophy and it has some basic rules or guidelines that should be met when using an Agile/Scrum style of project management [11]:

- Team size: 7 +/- 2
- Project/Product Owner: on site and intimately involved with Scrum team
- Daily Stand-ups: 1-3 15 minute meetings where team members inform on their progress, expectations, and potential problems
- Iterations: delivery of working components frequently and demonstrable
- Use of white boards and sticky tags for work-product progress
- Mutation of requirements over time with Project Owner able to change requirements often and sometimes in a significant manner.

While these practices of Agile/Scrum appear to be quite appropriate for software deliverables (bits), they do not seem appropriate when the deliverable requires a physical artifact or product prototype. For example, once the machinist has milled the spindle down to 4.3 cm +/- 0.5mm, a change in requirements to 4.8 cm +/- 0.5mm would require rework and waste accumulation. This is the point concerning the difference between the physicality of the software project and the physicality of the non-software project: bits can be altered without waste.

Strengths and Weaknesses of Agile

Agile style methodologies seem to have had some success in environments with common conditions:

- Software deliverables,
- Small team sizes (5-9 is usual sizes surveys contend) [12]
- Co-located deliverable's owner or owner's representative,
- Senior developers populating the team.

Beyond these four most common of discussed Agile/Scrum project characteristics, the success rates from the last Agile community survey from Scott Ambler [12] indicated that for 2009 the respondents said their Agile-inspired projects enjoyed a 10pt (58 vs 48%) improved success rate over the project management 'traditional' current best practices. [12]

There also appears to be some anecdotal evidence that Agile style methodologies have a statistically significant impact on projects that are of larger size and complexity [13] When confronted with the aspects of building something like the Colorado Hoover Dam, the Freedom Towers in NYC, the 'Big Dig' in Boston's Harbor, or again, the Millau Viaduct in France, Agile practitioners seem to be a bit less vocal about the benefits of Agile-style methodologies. For proof of these discussions, log on to any of the Agile Forums (Google for a current list), register and ask the following questions:

1. Please explain how Agile/Scrum would be used to manage a project that had as a deliverable the Hoover Dam?
2. Please explain how Agile/Scrum would be used to manage a project such as the ISS (International Space Station)?
3. Please explain how Agile/Scrum would deal with the legislative and legal requirements of documentation for a Phase III FDA clinical trial by a major pharmaceutical company?

Then sit back and let the community go to work; during the information annealing process that will take place, the range of answers and suggestions will most likely be both wide and broad. The answers will discuss having a Scrum of Scrum architecture where virtual teams are connected via their Scrum Masters that collectively form a Scrum Oversight Council. The SOC will then transfer the necessary information between Scrum teams there by allowing a 9 person SOC to coordinate up to 63 to 81 scrum team members depending on the size of the scrum teams.

The question that will remain open is how does Scrum deal with deliverables with an atom-based physicality? How are requirements firmed and agreed to when the requirements form the basis for the physical reality of the deliverable: ie., a computer chip's hardware logic diagrams and underlying mathematical profile, the formula for 1000 tons of quick-drying concrete that must be able to withstand underwater pressures of 2000 lbs per square foot, or even the welding parameters of a nuclear containment vessel? Three standups a day even with the owner in the room may not be enough to provide the documentation or requirement firmness a physical deliverable may demand.

Strengths and Weakness of Project Management CBP

Since this is the method with the 'home town advantage," this white paper will leave the discussion of the strengths and weaknesses of the project management current best practices (CBP) to the reader for obvious reasons of length and brevity. However, the strengths and weaknesses can be very quickly grouped into the following categories with the necessary details being discussed:

Strengths:

- Well-known processes with over 400,000 practitioners trained in their use,
- Significant number of successes in the physical deliverable arena,
- Support of very large project team with distributed locations and languages,
- Knowledge of what constitutes a successful completion (if planning done correctly),
- Arms-length relationship between project owner and project team.
- Up front estimation of project's scope, schedule, and budget, and
- Identified responsibility for the project's success or failure.

Weaknesses:

- Rigidity when applied to "bit-oriented" deliverables such as software projects,

- Limited ability to respond to scope changes during project execution,
- Overhead in definition of requirements, scope, and costs before anything happens,
- Limited interaction between project owner and project team members,
- Significant costs for project management activities, and
- Hierarchical sensitivity for political impact and influence.

While readers may have their own set of additions to these lists, the strength and weaknesses of the project management CBP can and do mirror their opposites in the Agile-style philosophies. This can explain why Agile is better suited to project/product deliverables of 'bits' while the CBP is of 'atoms.'

Comparison of Agile and PM CBP

Can this discussion be condensed down to a set of graphs that illustrate the areas of "applicability" for both Agile and CBP? Fortunately, the answer is Yes, and the following graphs are offered in support of this response. The first graph illustrates the comparison of Agile, CBP, and other project management philosophies and methods when plotting two independent variables:

1. The size of the project team as a proxy for size and complexity of the project, and
2. The proximity of the team and project/product owner.

These variable are independently correlated since the literature currently discussion both Agile-style project management philosophies and CBP-style management philosophies do not indicate that by choosing a team size automatically determines the proximity of the team and owner, or vice versa. From this graph, Figure 1, it can be seen that each philosophy and/or methodology has its area of 'applicability' as determined by the practitioners of the specific discipline. [14]

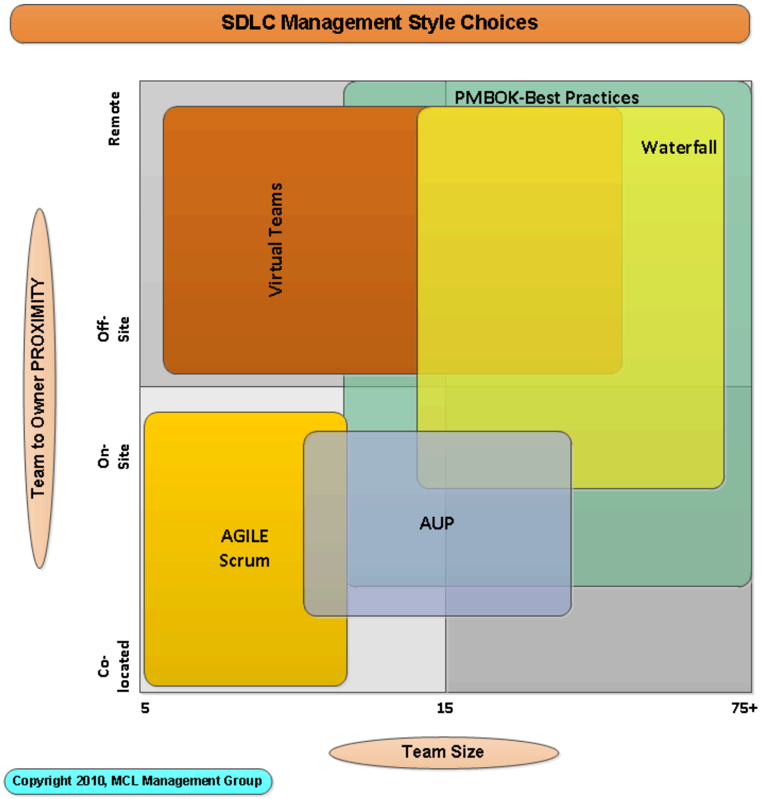


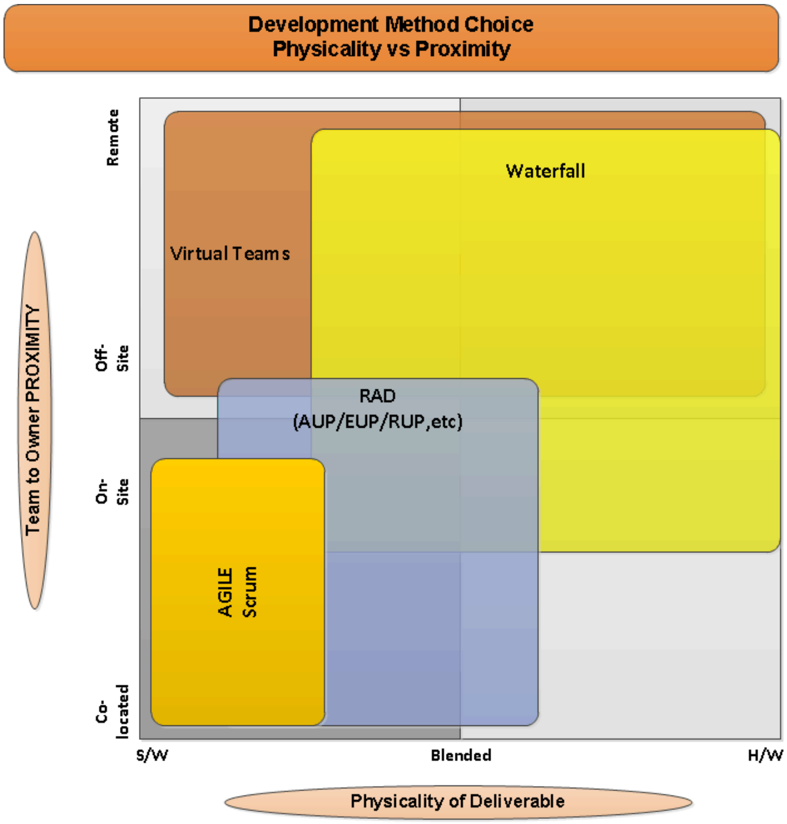
Figure 1

While there may be some argument over the exact placement of each philosophies' pad, using the documentation provided the current practitioners, these pads illustrate the best applicability of each of the graphed disciplines. What that reader should take away from this quadrant graph is that it shows why the application of project management CBP (the green area) just barely touches the applicability of the Agile-style methods. This would seem to show why Agile is enjoying such successes in the small team size within close proximity of the project owner while the more traditional project management CBP style methods had such a difficult time in providing the environment for success.

The reverse conclusion is also drawn that until the Agile-style philosophies and methods come to grips with distributed or remote located project owners and team sizes in the 25 to 75 range, and the current research is under way to attempt such solutions, the Agile-style methods will do better when paired with projects where these parameters are the project characteristics.

Major Differences between Agile and Project Management CBP

What are the major differences between the Agile-style philosophies and the project management CBP style? Can a graph again help to illustrate these concepts? The answer is again, Yes. Figure 2, The Area of Applicability based on the Physicality of the Deliverable, or 'Bits vs. Atoms' illustrates where these disciplines have their best impact. This can lead to understanding where practitioners can best deploy particular philosophies and their associated methodologies. [14]



Copyright 2010, MCL Management Group

Figure 2

The project manager or Scrum Master as each philosophy has differentiated roles and titles will have to decide if their particular philosophy is the correct choice given the particular characteristics of the deliverables they have agreed to offer to the project owner. Can each particular practitioner be counted to be objective enough to accept the limitations and boundaries of their particular form of project management?

This is one of the questions that each organization must seek the answer to before deciding on a particular philosophy opposed to all others. An analogy of myopic perspectives comes down to this quotation by a very observant practitioner of management philosophies:

> *"If all the dogmatic practitioner has in his toolbox is a hammer, then everything must look like a nail; if not, than he will apply his hammer until it resembles a nail."*
>
> CJ Stoneman, 1989.

Agreement and Points of Contention

Understanding the areas that Agile and CBP excel in given their particular strengths and weaknesses, an organization must decide if these two development philosophies can work in conjunction within the framework of the organization's expertise and human resource talents. It is precisely that Agile and CBP do not overlap in major applicability that they can be made to peacefully co-exist in their appropriate environments. However, if an organization has already become the single philosophy provider, it would be efficient to limit the choice of engagements and projects to those that are within the solution profile of the chosen philosophy.

To believe that Agile or CBP can handle any type of project shows a lack of acceptance that each philosophy has its appropriateness in solving a subset of project issues and problems depending primarily on the type of deliverable being requested of the project. Agile it has been shown is best applied to software deliverables where the lack of physicality and variation of requirements is handled via its light-weight approach to project execution. CBP can best be applied to those projects where the physicality of the project's deliverables demands a more formal, structured requirement definition and sign-off set of procedures.

Organizations depending on their size, scope of engagements, and desired professional image should be willing to limit their choice of projects to those where their expertise and skills are strongest in similar fashion to a professional project manager knowing their limitations based on experience and knowledge. For an organization to become 'co-opted' or dogmatically blinded as to the appropriate application of each development philosophy will in the end result in a troubled or misdirected project regardless of the passion of the project team. Remember the hammer analogy and maintain a more complete set of tools, not just a hammer.

Steps an Organization Should Take in Choosing a Development Philosophy

1. Know the strengths, skills, and experiences of your project teams,
   a. Ensure properly trained employees in your philosophy choices,
   b. Ensure a mixture of skill sets from beginners to senior practitioners,
2. Review your OPA (organizational process assets), and lessons learned for project expertise,
3. Review your marketing profile to ensure it aligns with your philosophy choices,
4. Review your engagement or project selection protocols for alignment with philosophy choice, and
5. Understand that one cannot be everything to everyone – know your limitations.

References

[1] The Agile Manifesto. http://www.agilemanifesto.org
[2] Wikipedia. "List of Software Development Philosophies."
http://en.wikipedia.org/wiki/List_of_software_development_philosophies

[3] Kent Beck. eXtreme Programming. http://www.extremeprogramming.org/
[4] Jeff Sutherland. Scrum. http://scrum.jeffsutherland.com/
[5] Scott Ambler. RUP, AUP, and EUP. http://www.ambysoft.com/
[6] Alistair Cockburn. Crystal. http://alistair.cockburn.us/
[7] Jim Highsmith. APM. http://www.jimhighsmith.com/
[8] The Standish Group. CHAOS surveys. http://standishgroup.com/
[9] Alistair Cockburn. DOI. http://alistair.cockburn.us/The+declaration+of+interdependence+for+modern+management+or+DOI
[10] Jeff Bezos. Two Pizza Rule. http://www.fastcompany.com/blog/fast-company-staff/fast-company-blog/sixteen-slices
[11] Jeff Sutherland. Scrum Papers. http://scrumtraininginstitute.com/library
[12] Scott Ambler. Agile Surveys. http://www.ambysoft.com/surveys/
[13] Agile Modeling. Scaling Agile. http://www.agilemodeling.com/essays/agileArchitecture.htm
[14] MCL Management Group. http://www.mclmanagementgroup.com

Authors' Biographies

For more information and project management philosophy-choice consulting, please contact the authors with your questions or comments:

Paul Lohnes, PMP phl_mp@mclmanagementgroup.com

Paul H. Lohnes has been active in project management for over 25 years beginning soon after he left the US Navy in 1981. After obtaining his BS (CS) and teaching as an adjunct at the UC Berkeley's Extension University for 6 years during which time he completed his MBA (Finance/Op Mgmt) at Golden Gate University, San Francisco, Paul started up a private consulting practice in project management of technical and computing projects. His clients over the years have included Fortune 100 companies in telecommunications, computing, networking, and finance in addition to developing and delivering over 500 technical and management seminars to over 10,000 attendees around the world.

Mr. Lohnes holds the PMI PMP certification, and is currently beginning the application process for the PMI's new risk management certification, the RMP. Mr. Lohnes is completing projects in the upstate New York area and returning to the MD/VA/WDC area for the purpose of starting a new company offering advanced project and risk management services to clients needing such components to their business management endeavors. Mr. Lohnes has developed several proprietary risk management and indexing tools that he uses in service of his clients and customers.

Ms. Cheryl A. Wilson, CCEP, PMP, PMI-RMP cawilson@cwpmanagementgroup.com

Ms. Wilson has an impressive resume and work history both in and around the Washington, DC area. She is a USAF veteran and a graduate of the University of Maryland with a Bachelor of Arts. Ms. Wilson is also a holder of a Masters Degree in Management Information Systems from Strayer University.

Being one of the first women to obtain the Risk Management Professional certification from the Project Management Institute, Ms. Wilson is in high demand both as a project risk consultant and risk analysis team leader. She has held several, high-profile project management and business analyst positions at firms contracting with the US Government.