

# The Art of Project Management – Book Review

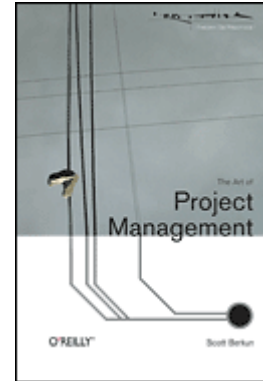
*By Craig Murphy*

**craig  
murphy.com**

## The Art of Project Management

Author: Scott Berkun  
Published Date: May 05  
Price: £28.50 (RRP)  
Length: 500 pages  
ISBN: 0-596-00786-8

<http://www.oreilly.com/catalog/artprojectmgmt/index.html>



### Introduction

Weighing in at a little under 500 pages, sitting at an inch thick, The Art of Project Management makes for a good travelling companion. Its sixteen chapters are split into three sections: Plans, Skills and Management, each containing five chapters.

If you think that this is just another book about project management, I urge you to read on. This book is about project management, but you won't find any project plans, PERT charts, Gantt charts or any form of methodology-specific advice.

What you will find is a very readable and well structured collection of chapters that offer you material that generally doesn't make it into books about project management. I'm talking about the "soft" skills that a project manager needs in order to make progress. But these aren't just skills that project managers need, anybody involved in the project needs them too. So please, read on and learn how you can become a better citizen in the project(s) that you are involved with.

# The Art of Project Management – Book Review

## Chapter Titles

1. A brief history of project management (and why you should care)

### I. Plans

2. The truth about schedules
3. How to figure out what to do
4. Writing the good vision
5. Where ideas come from
6. What to do with ideas once you have them

### II. Skills

7. Writing good specifications
8. How to make good decisions
9. Communication and relationships
10. How not to annoy people: process, email, and meetings
11. What to do when things go wrong

### III. Management

12. Why leadership is based on trust
13. How to make things happen
14. Middle-game strategy
15. End-game strategy
16. Power and politics

# The Art of Project Management – Book Review

## What's in the Chapters?

### Chapter 1: A brief history of project management

You shouldn't be surprised to learn that this chapter positions project management using a variety of historical references stemming back to the pyramids of ancient Egypt and reaching forward to modern-day projects such as the Hubble telescope and the Boeing 777. Clearly we can learn from successful projects past and present.

The role of project management in general is covered as is the role of project management with Microsoft. Since Berkun is himself a former Microsoft employee, it is fair to say that we're going to be reading about how projects are managed within Microsoft and hear all about their war stories. We're all familiar with a lot of Microsoft products, so why not let the author tell us how they came to be on our desktops?

Early reference to Tom Peters work (Pursuing the Perfect Project Manager) sets the scene for the rest of the book. We are treated to a plethora of quotes and references to de facto authors and publications. You are encouraged (by me at least!) to follow up these references if you wish to learn more. Berkun makes use of other authors' work in order to make his point and keep his page count down.

As a project manager, it's a balancing act that involves being able to combine traits, skills and attributes whilst carefully factoring in pressure and distraction. Getting the mix between process, goals and active involvement is important if project managers are to create unique value. Berkun tackles these last two sentences with a simple checklist based on Tom Peters' original work. He adds weight to his discussion with reference to Fred Brooks (The Mythical Man-Month).

This chapter's summary left its mark on me: "If you keep a beginner's mind, you'll have more opportunities to learn". How true.

### Chapter 2: The truth about schedules

Berkun opens with the sentence: "People tend to be late". He follows this with recognition that tardiness doesn't limit itself to private/social lives, it makes its way into the commercial/corporate environment too. Berkun goes on to discuss the need for scheduling and the importance of being sceptical during the initial schedule creation (as opposed to being optimistic which seems to be the accepted norm).

Early mention of 'agile' and eXtreme Programming (XP) caught my attention. It's good to see agile and XP making it into what could be considered the mainstream publishing market. Similarly, Berkun is quick to point out that risks should be explored as early in a project as possible.

Berkun's writing approach is open and frank – he does not write ten words when only two will do. His honesty shines through and were it not for this, I believe that this book would have been twice as thick. I was pleased to see a great number of "thoughts that I usually keep to myself" actually written down such as: *"Programmers should be trusted": if your brain surgeon told you the operation you need takes five hours, would you pressure him to do it in three?*

It is necessary to obtain project team "buy-in" to schedules. Berkun recognises that this does not happen often and includes "ownership and commitment" in a list of common oversights. Interestingly, some of the oversights are items I have previously been asked to remove from my schedules (such as sick time). Checklists play a major part in this book – typically each checklist item is described using a bullet point and a full paragraph of explanation. You are encouraged to adopt and adapt the checklists in your own work.

This isn't a chapter that teaches you how to prepare a detailed schedule. Instead, it's a chapter about what purpose schedules serve, what they look like and why schedules so often

## The Art of Project Management – Book Review

fail. You won't find a PERT chart or a Gantt chart in this chapter, something that this reviewer finds refreshing. What you will find however, are simple "hand-crafted" (electronic, but made to look that way) diagrams lightly populating this chapter. Like the remainder of the book, there is a lot of reading, however Berkun's writing style is gentle, well laid out, and contains a mix of small case studies that set the scene for the surrounding paragraphs.

### Chapter 3: How to figure out what to do

Software planning is an emotive topic. Berkun controls the emotion with the early introduction of a lengthy quotation from software planner extraordinaire, Fred Brooks (The Mythical Man-Month). This theme, the provision of a quotation, stands true for all the chapters and provides a good "scene setter" that usually acts as encouragement to read the rest of the chapter.

We are presented with four typical planning deliverables: a marketing requirements document (MRD), a vision/scope document, specifications and a work breakdown structure (WBS). The first three are likely to be very analogous to artefacts you may have in your organisation. The WBS means something different to me; Berkun uses it to record a prioritised list of work items, who will work on the work items, how the work items can be tracked, etc.

This chapter's key learning point stems from a good discussion of a project's three primary drivers: technology, business and customer. I was pleased to see a discussion that gives the customer at least equal weighting – all too often their involvement in a project ends after the requirements have been gathered. Each of the primary drivers is explained and is augmented with the author's own experience. Each of the primary drivers enjoys a checklist of important questions that often go unasked or are answered using assumptions rather than reality – my favourite is: *"What do people actually do? (Not what we think they do or what they say they do.)"*

Berkun presents us with a "catalogue of common bad ways to decide what to do", all of which are drawn from his personal experience. Whilst these are eminently familiar, they are slightly tongue-in-cheek (but they do help firm home the point).

It's worth noting that this author likes to keep things simple and breaks product development into four phases: initial planning, design, programming and testing. That's not to say that the content of this book is aimed directly at traditional or waterfall project management approaches, even agile and iterative development approaches incorporate these four phases across iterations.

The "planning phase" is only complete once all of the planning deliverables (MRD, vision, specifications and WBS) are complete. A "waterfall" model is used to document the interaction between each of these. Initially, interaction between each of these deliverables is permitted. As time goes by, Berkun suggests that interaction is limited to the specifications, i.e. activities that are near the bottom of the waterfall.

Once planning is complete, this chapter focuses on how requirements can be brought together, problems are turned into solutions/scenarios and feature lists are drawn up to form a vision document.

This chapter makes reference to other seminal works, Exploring Requirements: Quality Before Design (Weinberg & Gause) and Writing Effective Use Cases (Cockburn) demonstrates that this author wishes to keep his text focused and meaningful. Over the course of this book, some 90 outside references are made; these are brought together in a "Notes" section. When I receive a new book, I often go straight to this section – a lot can be gleaned from the work an author makes reference to. I'm pleased to report that Berkun makes reference to some of the best authors from a wide range of industries, not just software and not just project management.

# The Art of Project Management – Book Review

## Chapter 4: Writing the good vision

The five key attributes of a good vision document are covered: Simplifying, Clear Intent, Consolidated, Inspirational and Memorable. These attributes should largely be self-explanatory; indeed the last two should also be eye-openers for most readers...can you remember the last time you read an inspirational vision document? That memorable? These are five good attributes that the author explains in a succinct fashion, free from any waffle.

This chapter goes on to provide an excellent list of key points that a vision document should cover, presented as a list of questions.

Berkun makes an excellent statement: “The project manager should seek out the smartest and most sceptical members of the team, and enlist them to poke holes in the logic and supporting arguments behind key statements.” The easy route isn’t an option: the smart sceptics are folks most project managers shy away from because they create work instead of facilitate work: sometimes grabbing the bull by the horns is the better option.

I was glad to see a few words about “On writing well”. It is hard to be simple (in writing), it does require one primary writer (to maintain style throughout) and volume is not quality. Think back to the last vision document you read/wrote, did it satisfy the five key attributes mentioned previously? How many pages long was it?

A “catalogue of lame vision statements” provides us with four clear examples of vision statements that are clearly rather poor. Again, these are somewhat tongue-in-check, but they do give clear advice on what not to do. This chapter wraps up with a few examples of good visions and goals. The key takeaway is that single-sentence visions should be to the point, be goal-driven and be achievable.

## Chapter 5: Where ideas come from

Berkun is a firm believer that requirements (of high quality) help avoid mistakes. He backs this up with the provision of five common mistakes. Whilst not stated implicitly, I felt these common mistakes actually form a “refinement iteration” that harvests inspiration and leads to clearer requirements.

Given a good set of requirements, design exploration takes us through what makes up a good idea and what makes up a bad one. Berkun has little good to say about the phrase: “there are no bad ideas”. By “bad idea”, he really means the “awful, horrible, useless, comically stupid and embarrassingly bad ideas”...it’s all about definition: I choose to define a bad idea as one that might solve the problem in hand, but one that might cause further problems downstream.

Luckily, Berkun realises that given a collection of ideas it is difficult to weed out the good from the bad, particularly as the value of an idea is very difficult to judge. Further, he recognises that sometimes we have to spend time with bad ideas when he writes about “Bad ideas lead to good ideas”. Indeed, there is clear recognition that making mistakes and working with bad ideas can help us arrive at a good idea. As IBM’s Tom Watson once said: “If you want to succeed, double your failure rate.”

“Good questions attract good ideas” reminded me of my days on a helpdesk – ask the right (good) questions and even the most non-technical person will provide good answers that leads to his/her problem being solved much sooner. To help us ask/identify better questions, Berkun discusses how we can ask more focusing, creative and rhetorical questions.

## Chapter 6: What to do with ideas once you have them

So far, The Art of Project Management has been an excellent read. Sadly, with this chapter, I struggled to relate to the examples and the narrative. That said, this chapter does have a good discussion about the use of prototypes. Recognising that “design is an exploration”, sometimes the only way to move exploration forward is to promote a design to a prototype.

## The Art of Project Management – Book Review

Berkun explains where prototypes start (picking a candidate design or idea) and prototyping with and without user interfaces. Berkun explains that prototypes serve to reduce risk and surface new risks, two things that can help a project and a project manager. I was pleased to see Berkun promote prototypes as being good for the programmers – prototypes convey an indication that a design has been thought through and hasn't just been picked at random from a set of viable designs.

This chapter closes with a look at open-issues lists. I was pleased to learn that Berkun uses "visible open-issues lists", typically using a white-board. Making project information highly visible is a useful trick to adopt. Firstly, the whole project team can see where progress is being made and where it's not. Secondly, those folks who stop by looking for a progress update can be directed to the whiteboard. It can be a great timesaver! Surfacing a project's "data" and progress is something all projects should do, there should be no secrecy, honesty is key.

### Chapter 7: Writing good specifications

According to Berkun, specifications should do three things for a project. Specifications should: ensure that the right product gets built, provide a schedule milestone and enable review/feedback. Like most chapters, a useful checklist is provided – in this case a checklist of what specifications can and cannot do. I was suitably impressed with the frankness proffered by this author: specifications are frequently used to reduce or eliminate discussions (as in "you've got a specification, why do you need another discussion?")

Being methodology-agnostic, this book has to home in on the key points that make a good specification. It is recognised that specifications must incorporate and consolidate all requirements, feature specifications, technical specifications, work-list items, test criteria and milestone exit criteria.

Simplification of specifications is an important point, and one this author makes when by the provision of seven specification writing tips. My favourite tip was: "Don't fall in love with Visio or flowcharts". Sometimes we can overuse a diagramming tool to the extent that it actually clutters a specification. Berkun's Microsoft influence shines through in this chapter, even to the point that what used to be Microsoft-specific jargon "spec complete" (hence McConnell's book Code Complete) appear in this chapter.

### Chapter 8: How to make good decisions

This is a subject close to my heart. Over the course of 24 pages, a decision is dissected revealing a complex beast, and a beast that Berkun teaches us to tame to our advantage. From the point where a decision becomes a meta-decision, i.e. a decision that affects one or more sub-decisions, through to the point where previous decisions are examined thus revealing better ways of doing things, this chapter guides us carefully through the lifetime of a decision.

### Chapter 9: Communication and relationships

This is an excellent chapter that deals with two key facets we all should pay attention to, not just those of us in a project management role. Without good communication and solid relationships, we are lost.

Berkun present us with a basic model of communication whereby e-mail/information moves through five states: transmitted, received, understood, agreed and converted to useful action.

Seven common communication problems are noted and discussed, each problem receiving one third of a page. The problems covered are: assumption, lack of clarity, not listening, dictation, problem mismatch, personal attacks and derision/ridicule/blame. A very salient quote is appears during the discussion about "not listening" – it sums up the need for good communication and good relationships: "*They actually listen to me, instead of just waiting for their next chance to talk*".

## The Art of Project Management – Book Review

That last quote came from the film Fight Club – this author’s interjection of salient quotes throughout each chapter keeps the book eminently readable and ensures that the reader does not tire.

I was delighted to see a good discussion about “how to get people’s best work”, and it got off to a good start: “Good leaders rarely force people to do anything”. A further eight points go on to provide excellent advice on how to elicit good work. Related to the “not listening” communication problem noted earlier, I was glad to see “follow advice” listed. The easiest way to get folks to do good work is to find out what’s bugging them and solicit their advice/solution...then implement it.

This chapter concludes with a look at the subject of motivation and it makes use of my favourite phrase: “*move the project forward*”. A chapter covering communications, relationships and motivation, you couldn’t ask for more.

### Chapter 10: How not to annoy people: process, e-mail and meetings

This is my kind of chapter. I’m sure that we can all remember reading an e-mail that either frustrated or annoyed us – but why did it cause an annoyance? This chapter provides five summary points answering this question. According to Berkun, we become annoyed because other folks assume we’re idiots, don’t trust us, waste our time, manage us without respect or make us listen to or read stupid things. Well, that hits the nail on the head.

Berkun then goes on to discuss the subject of process. I was impressed with brashness and terseness used in this chapter: “any idiot with enough authority can come up with the most mind-numbingly idiotic system”. Why mix words and why write paragraph after paragraph explaining the effects a bad process can have on a project team?

I was similarly impressed with the acceptance that what worked in the past should work now in the present. All scenarios are different and it is important to consider the new scenario fully, before rolling out an old process: “emphasise the needs of the present over the observations about the past”. Nicely put.

This chapter goes on to present examples of good and bad e-mail messages. It laments about the fact we all don’t write good e-mail all of the time and offers simple advice: be concise, simple, direct, offer an action and a deadline, prioritise, don’t assume people read anything, avoid sequences of events and avoid FYIs. But, and this is very important, Berkun’s last piece of advice is to use the telephone – well said, nothing is better than a conversation.

Meetings are the bane of most projects – this chapter finishes off with an examination of how best to run a meeting. Berkun’s “meeting pointers” wrap up this chapter nicely. Mention of Scrum, stand-up meetings and Scrum’s three questions (What have you done since the last meeting? What are you going to do before the next meeting? What is blocking you?) was a pleasant and welcome surprise.

### Chapter 11: What to do when things go wrong

“No matter what you do, things will go wrong”. Accept this fact and you’re well on the way to becoming a good project manager. How you deal with something going wrong is the subject of this chapter.

In keeping with other chapters, a checklist offering advice on how to deal with “when it goes wrong” is presented. One of the checklist items caught my attention, or rather its footnote did. During a discussion about “calming down”, Berkun recognises that humans need to express their emotions. When something goes wrong in a project, taking the time out to work out those emotions is very important (perhaps you use a shoot’em up to iron out emotion and tension?)

This chapter goes on to present a further checklist outlining the situations that cause projects to go wrong in some way. Once something has gone wrong, the subjects of damage control,

## The Art of Project Management – Book Review

conflict resolution and negotiation are discussed. These might sound like obvious topics to discuss in a chapter like this however Berkun brings these topics to life with good use of examples and reference to other authors.

The remainder of this chapter presents an “emotional toolkit”, which might sound like an odd thing to include in a book about [the art of] project management. This is what sets this book apart from other traditional project management books: it dares to be different and it includes material more commonly found in self-help books. That should tell you something about this book: it’s not just about project management, it’s about people management – and without good people, projects fail.

The emotional toolkit has a discussion about pressure and explains why we all need the right kind of pressure in order to make progress. It also covers feelings and presents a brief discussion about how a project manager should learn to recognise that other people’s emotions may not be directly related to the actions you have taken. Wise words, the importance of understanding “the bigger picture” springs to mind, i.e. don’t consider feelings in isolation.

### Chapter 12: Why leadership is based on trust

Twelve chapters in and I am becoming more and more enlightened by Berkun’s dictionary definitions at the start of most chapters (“trust” for this chapter), and the third party quote that follows it: *“Trust is at the core of all meaningful relationships. Without trust there can be no giving, no bonding, no risk-taking”*.

Over the course of this chapter, we are taught that trust is built through commitment and that it is lost through inconsistent behaviour. We also learn that trust should be defined; employees should be given a level of authority to make decisions, i.e. trust and authority should be delegated.

The rest of this chapter provides a good discussion about trust, what to do when people make mistakes and feedback processes. I was particularly interested in Berkun’s statement that “trust is insurance against adversity”. Getting people to work with others in a cooperative fashion is important, if the environment is lacking in trust, people will work in spite of each other.

### Chapter 13: How to make things happen

What skills do project managers have or need in order to make a project run smoothly? The answer is simple: “the ability to make things happen”. We can argue all we like about the other skills we think a project manager needs, but it all boils down to this one key skill. That said, how we make things happen is the subject of this chapter and one in which we find some of this author’s key insights.

This chapter opens with a discussion about priorities and lists. Berkun presents us with the notion that “list maintenance” is something that helps us get things done. Whilst this outlook is possibly simplistic, I had my eyes opened when I sat down and thought about my most recent project and what I did most of. Oddly enough, I spent a lot of time juggling lists of prioritised work items. Considerable emphasis is put on prioritised work items (particularly priority 1 items), and rightly so: they help focus the mind and thus drive the project forward, a point not missed by this author.

We all struggle to say “no”. Don’t fool yourself, we do struggle. When you come out of denial, you’ll be pleased to be treated to some good common sense ways of saying no, such that it is for the good of the project. You may already have mastered the art of saying “no”, but if you haven’t, Berkun provides us with a list of the main methods of saying “NO!” along with good examples of how to apply the method.

Given that this is Berkun’s first book, I was impressed to see that he wasn’t afraid to use real blasphemous English. For example, on page 343 we find “Calling bullshit makes things



## The Art of Project Management – Book Review

happen”. Sometimes you just have to tell it how it is, or in this author’s lingo: “keeping it real”. Maintaining a sense of reality on a project is important,

Under the heading of Guerrilla Tactics, the notion of being savvy is taken to the next level. We are presented with eleven tactics that can be adopted in order to help get things done or make things happen. Some of these are enlightening, and you should buy this book if you want to learn more. One that you might be practicing already or at least be aware of is the tactic of getting out of the office, away from your desk, working from home, or as Berkun puts it: “hide”. Hiding is becoming much easier these days, especially with the power of wireless networking – hiding in famous coffee outlets is a reality!

### Chapter 14: Middle-game strategy

This chapter and the next take their titles from the game of chess. Here we are 358 pages in and now we start to discuss tactical project management. What came before this was a great discussion about the softer side of project management, the people skills. Now that the word “game” has entered the equation, we can be sure that this chapter (and the next) are going to cover how best to play the project management game. And that’s what it is, a game that consists of a series of well thought out moves taking us from project opening to project completion.

With the briefest reference to C programmers and the assert() method, this chapter focuses on sanity checks. It presents us with a list of questions that we should ask ourselves daily, weekly and monthly – each with a view to keeping the project on track and ensuring that you as project manager make the right moves.

Middle-game work revolves around planning and the appropriate sequencing of task or work item allocations, or pipelining according to this author. Management of change also figures in this chapter with the author borrowing the commonly used design change request (DCR) terminology.

### Chapter 15: End-game strategy

This chapter presents a very neat analogy that the author uses to help us understand how to drive a project to finish on time. The analogy is that of landing an aircraft. Landing an aircraft requires a smooth glide path, which when graphed appears to be a straight line. In reality, it is a series of very small manoeuvres, just like a project moving from a start date to an end date.

Charts and other visibility indicators play a major part of this chapter. What I like about Berkun’s charts and graphics is their simplicity. Folks don’t like or understand complicated charts, they prefer the information presented in the chart to jump out at them. A lot of projects would enjoy morale gains, performance gains and schedule gains if Berkun’s charts were adopted.

Interestingly, mention of eXtreme Programming (XP) again makes it into this chapter, with a reference to the customer becoming more involved. Rapid feedback is important, and who better to provide that feedback than the customer, the final arbiter in the equation. Berkun rightly notes that bringing a customer into a project for the first time can be difficult but does emphasise that it has massive pay offs downstream.

### Chapter 16: Power and politics

So, you’re not political? I have news for you: project management involves politics. No matter how hard you try to avoid it, you will ultimately become embroiled in a political situation regarding your project. It’s almost like life and death: a certainty. And it is this reason why Berkun chooses to devote an entire chapter, some 30 pages, to the topic.

Politics revolves around power and the misuse of power. Understanding where power (in a project) comes from can help us learn how to harness it and use it to the project’s advantage. Berkun refers to Thomas Quick’s Power Plays, noting the sources of power mentioned there.

## The Art of Project Management – Book Review

He then goes on to discuss how power can be misused, i.e. to the detriment of the project. Good use of Venn diagrams helps to convey the importance of finding the sweet spot or spots that highlight the areas of power that are good for the project (i.e. the overlap between “what’s good for me/my team”, “what’s best for the project” and “what best for team X”).

If ever you were stuck for solutions to your political problems, Berkun’s final section provides you with some 12 pages of advice. You probably already know that you need more resources, more authority, more influence, adjustment of goals and advice, this last section talks you through the tactics required to alleviate your political problems.

### Should you buy this book?

You should buy (read and implement!) this book if you wish to become a project manager or are an existing project manager wishing to become better or if you wish to take on a role that involves working with people. It is a book littered with common sense that will help move your projects forward and move your relationships with your direct reports and your family/friends to new levels.

If you enjoy reading books that are aligned with your way of thinking, yet still they manage to provide you with new and improved ways of doing things, this book is for you. Similarly, if you’re wondering what project management is all about, this book should be your first port of call.

### Conclusions

This is Berkun’s first book and with it he joins a short list of cult authors including, but not limited to: Steve McConnell (Code Complete, Rapid Development; After the Gold Rush), Frederick P. Brooks (The Mythical Man-Month) and Jim McCarthy (Software Project Survival Guide, Debugging the Development Process, Dynamics of Software Development).

When recruiting new staff I often ask them the question: "Have you read McConnell's work?" Those interviewees who do not know who McConnell is, or cannot even name the books are removed from the shortlist. I would like to think that one year from now I will be able to ask: "Have you read The Art of Project Management?"

Finally, we have a book that doesn’t just teach project management, it teaches the art of project management – this book teaches you the people skills that are required to bring projects in successfully. More importantly, it is a portable book of common sense – it should be required reading for those folks on a project and anybody who has interaction with the project team, e.g. upper management, sponsors, CEOs, CIOs, information providers and even customers.

### Overall Book Rating



[5 / 5: Must Read]

I don’t normally rate anything 5/5 – mainly because it suggests that there is little more than can be done to improve the work being rated. This book isn’t without a few niggles, however, I struggle to find major fault with it.

### Useful Resources

Scott Berkun’s web-site: [www.scottberkun.com](http://www.scottberkun.com)

Sample chapter: Chapter 3: How to figure out what to do  
<http://www.oreilly.com/catalog/artprojectmgmt/chapter/ch03.pdf>

# The Art of Project Management – Book Review

## About the Reviewer

Craig writes approximately fifteen articles per year, over 50 of which have been published. In addition, he has prepared and delivered over 40 presentations at developer conferences and user group meetings throughout the UK. All of the source code and PowerPoint slides are available from his web-site: <http://www.craigmurphy.com>

He is a regular contributor to [The Delphi Magazine](#), the [UK Borland User Group](#) magazines and has written for [ASPToday](#), [International Developer](#) and Computer Headline.

Craig is a founder member of [Scottish Developers](#) and works closely with [Agile Scotland](#). He played a major part in the organisation of the UK's first [DeveloperDeveloperDeveloper](#) event.

In particular, he will speak about SOAP, web services, XML, XML Schema, XML Topic Maps, XSLT, Delphi, and C#. He will also speak and write about Test-Driven Development (TDD) and eXtreme Programming (XP). Craig is a Certified ScrumMaster (CSM), Microsoft Most Valuable Professional ([MVP XML Web Services](#)) and holds an Honours degree.

<http://www.craigmurphy.com/blog>

[craig.artofpm@craigmurphy.com](mailto:craig.artofpm@craigmurphy.com)



It's all about  
Community  
Community  
Community

## First Published

24<sup>th</sup> June 2005

s c o t t i s h  
DEVELOPERS

<http://www.scottishdevelopers.com>