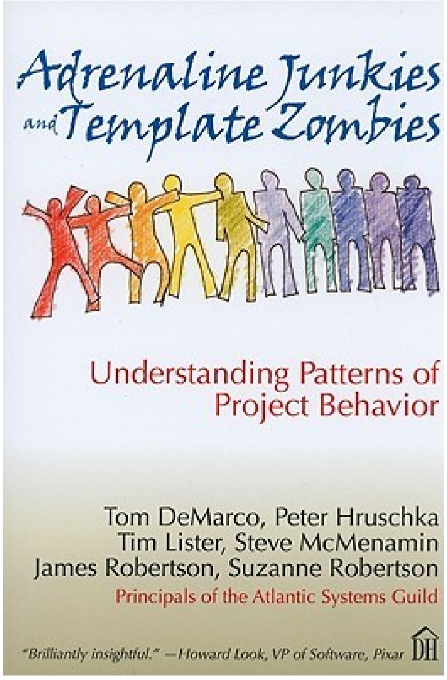

Adrenaline Junkies and Template Zombies
Understanding Patterns of Project Behavior
Tom DeMarco, Peter Hruschka
Tim Lister, Steve McMenamin
James Robertson, Suzanne Robertson
Principals of the Atlantic Systems Guild
"Brilliantly insightful." —Howard Look, VP of Software, Pixar
DH

Adrenaline Junkies and Template Zombies: Understanding Patterns of Project Behavior

Tom DeMarco , Peter Hruschka , Suzanne Robertson , Timothy R. Lister

Adrenaline Junkies and Template Zombies: Understanding Patterns of Project Behavior

Tom DeMarco , Peter Hruschka , Suzanne Robertson , Timothy R. Lister

Adrenaline Junkies and Template Zombies: Understanding Patterns of Project Behavior Tom DeMarco , Peter Hruschka , Suzanne Robertson , Timothy R. Lister
Adrenaline junkies, dead fish, project sluts, true believers, Lewis and Clark, template zombies . . .

Most developers, testers, and managers on IT projects are pretty good at recognizing patterns of behavior and gut-level hunches, as in, I sense that this project is headed for disaster.

But it has always been more difficult to transform these patterns and hunches into a usable form, something a team can debate, refine, and use. Until now.

In Adrenaline Junkies and Template Zombies, the six principal consultants of The Atlantic Systems Guild present the patterns of behavior they most often observe at the dozens of IT firms they transform each year, around the world.

The result is a quick-read guide to identifying nearly ninety typical scenarios, drawing on a combined one-hundred-and-fifty years of project management experience. Project by project, you'll improve the accuracy of your hunches and your ability to act on them.

The patterns are presented in an easy-reference format, with names designed to ease communication with your teammates. In just a few words, you can describe what's happening on your project. Citing the patterns of behavior can help you quickly move those above and below you to the next step on your project. You'll find classic patterns such as these:

* News Improvement
* Management By Mood Ring
* Piling On
* Rattle Yer Dags
* Natural Authority
* Food++
* Fridge Door
* and more than eighty more!

Not every pattern will be evident in your organization, and not every pattern is necessarily good or bad. However, you'll find many patterns that will apply to your current and future assignments, even in the most ambiguous circumstances. When you assess your situation and follow your next hunch, you'll have the collective wisdom of six world-class consultants at your side.

Adrenaline Junkies and Template Zombies: Understanding Patterns of Project Behavior Details

Date	:	Published March 1st 2008 by Dorset House Publishing Company, Incorporated (first published January 1st 2007)

ISBN	:	9780932633675
Author	:	Tom DeMarco , Peter Hruschka , Suzanne Robertson , Timothy R. Lister
Format	:	Paperback 238 pages
Genre	:	Business, Management, Nonfiction, Computer Science, Programming, Science

Download and Read Free Online Adrenaline Junkies and Template Zombies: Understanding Patterns of Project Behavior Tom DeMarco , Peter Hruschka , Suzanne Robertson , Timothy R. Lister

From Reader Review Adrenaline Junkies and Template Zombies: Understanding Patterns of Project Behavior for online ebook

Flavius says

A fair book. If you've been in software development for a few years, most of the stuff the authors write about will seem very familiar. The book's real value is given by the very clever pattern names.

A few examples:
- Nanny: the (project) manager who nurtures the team and helps them develop
- True believes: a zealot of a certain school of thought
- Project sluts: taking on more work than available capacity
- Undivided attention: people work only on one project.

Wendy says

I really want to give all my coworkers copies of this book. Because the coolest thing about it is that it gives you a language to talk about software development projects with. This book is basically a dictionary of "project patterns" - patterns of behavior, good and bad, that software development groups often exhibit.

Anyone who has worked in software (or probably on other complex projects) is going to recognize many of these patterns. (Although, I'm personally very grateful that I don't recognize some of the really dysfuntional ones.) Since reading this book, I find myself wanting to say, "Hey, I smell a dead fish," when we're talking about a schedule that we all know we can't meet, but no one wants to say anything. Or, "Wow, that new hire is totally Ben," when I'm trying to convey that she clearly loves her job and makes the workplace better for everyone around her.

You'll pick up some practical tips and advice from this book as well, but by far the most valuable thing is being able to describe stuff that you've always known about, but never had words for.

Patrick says

As far as patterns books go, this one was pretty light on the "pattern" part. It was more a series of short essays about things observed on project teams. Some of them offered advice. Some of them were more anti-patterns--things to avoid. Some were just strange. (The "IT seems to have a higher proportion of musicians" was a particularly useless 'pattern'.)

There are a few gems in here, especially around distributed development and high performing teams. There was lots that will be familiar to Agile practitioners, and there was lots that you'll recognize as happening in your own organization, both good and ill.

Overall, this is an easy read with some value, but if you're trying to learn about project management, you'd be better off starting with more focused books. On the other hand, if you're currently running a team or three, it couldn't hurt to leaf through here to remind yourself what you're doing right--and what you need to fix.

Roman Chumakov says

????? ??? ?????? ?????? ???????? ?? ???? ?????. ? ????? ????????? ????? ????????? ??????? ????? ??????. =) ????? ?????? ??????? ?????????????? ? ?? ???-?? ????????, ?? ??????? ??? ??? ???????? ? ? ????? ???????????? ?????.

Mircea Nistor says

It's funny. Some of the stories (typologies) are no longer actual but most are and you should be afraid that you might find yourself in one of the not so nice ones.

Denis Vasilev says

???????? ????? ?? ????????? ? ?????????? ???????? ????????? ?????? ?????????????. ?????????, ???????, ?????????.

Will says

It's all right.

The problem that this book has is that it's a list of patterns. Each of these patterns stands on its own, but does not feed into an overarching theme. As such, I got the feeling of "yes, I've seen this before" but nothing that told me "ah, this is why this happens." Simply listing things isn't going to lead to any epiphanies, but it's nice to see that some behaviors seem to be universal.

Tati says

Love this one!

David Lindelof says

Reading this book felt a bit like going to church to listen to the nth version of the same sermon. You agree with everything that's being said, you are comforted in your own beliefs, and you would certainly give this book to a newcomer to the field.

That said, anyone who's been through Peopleware or similar books will probably not learn any really new things from this book. It's very fun to read about those behavioral patterns that the authors have documented, but I would be much more interested in reading about patterns that work, rather than anti-patterns. And this book is 90% anti-patterns.

It's definitely a good, well-researched book, but I would prefer Coplien's "Organizational Patterns of Agile Software Development" book instead.

Ben Linders says

The book "Adrenaline Junkies and Template Zombies, Understanding Patterns of project behavior" contains 86 patterns, written in an easy readable and recognizable style which make you think about how people behave in projects. The authors have a lot of experience in various areas; their different views help you to get a broader understanding of issues that you can face in projects.

Patterns help us to understand things. They are not necessarily good or bad; in fact they may or may not be applicable given the situation at hand. I see many of the patterns my projects and daily work. Reading the patterns made me think on how we do things, and helped me to discuss them with colleagues. Below some examples of patterns, and an impression on how they helped me to understand and improve my way of working.

37: Talk then write: This pattern states that decisions made in meeting should be immediately afterwards communicated in writing. Pretty obvious, but I still see it happening that it takes a long time before we communicate what we decided, or that it is not communicated at all, that we do not inform all involved, etc. Due to time, the decision loses force, becomes unclear, forgotten, or it may even create resistance. With agile we have much better means now to communicate instantly, so let's do it! Use the war rooms, information radiators, wiki's etc, as much as possible, quickly after decisions are made, and update them when decisions are revised.

56: Undivided attention: This pattern states that undivided attention in a single project improves individual performance. With agile we usually have technical teams with full time members, focusing on the iteration at hand. But what about supporting functions? They are usually allocated to multiple projects, and also in process support, improvements, line support, etc. Task switching makes it difficult to give sufficient attention to each assignment, to build up relationships with the teams, and to really understand the issues and provide effective and timely solutions. Maybe it would be better to combine some of those part-time roles in a project, and have somebody working most of his/her time for one project? Or make the business case for a support role, increase the funding and assure that somebody can spend at least half time on one project, in stead of just of couple of hours or a day each week.

These are just 2 of the many patterns from the book that made me think about things. Reading the book made me feel like attending a good conference. There are a lot of interesting subjects in there, some which confirm what you already know, some which are less relevant for you, but there are certainly useful things in there. So most probably you will find some patterns that help to think about your work, to discuss it, and to improve it.

Michael Coté says

As with most pattern books, this is one you flip through in an hour and then save it to refer back to. The strength of software management and development pattern books is describing problems that commonly occur, not really telling you how to fix them. Thus, they tend to be frustrating because you're left thinking, "how am I going to get this to work in my organization?" There is a certain level of detail in some of these patterns that's refreshing, but most are just brief outlines of a software management or team-work problem.

Still, they're extremely helpful things to keep in mind which you may be forgetting ("The Empty Chair") or no longer think applies to you ("Young Pups Old Dogs"), helpful advice if you must do it ("Offshore Follies"), to some that can be reduced to a clever quip, as in "War Room" where DeMarco says, "I'm beginning to think that a project not worth a war room may be a project not worth doing."

There's solid advice in here, but the 0th pattern is "Be humble: never assume you have this shit figured out." After that, many of them are extremely good advice.

Valery Dostovalova says

???????? ?????. ??? ??????????? ????????????, ??? ???????? ??? ?? ? ???? ???????? :)

Johnny Graber says

Adrenalin Junkies & Formular Zombies ist ein sehr gut gelungenes Buch über die verschiedenen Muster die einem bei Projekten immer wieder begegnen. Die teils lustigen, teils aber auch sehr tragisch Situationen werden im Buch eingehend beschreiben und mit Warnhinweisen versehen. Transparenz ist nicht immer eine gute Strategie und viele Firmen bevorzugen die traurige Wahrheit erst zu einem Zeitpunkt zu erfahren, wo jegliche Chance auf eine Verbesserung verpasst wurde.

Sehr gefallen hat mir das Muster mit der blauen Zone. Diese liegt zwischen der grünen Zone (wofür man die explizite Erlaubnis hat) und der roten Zone, deren betreten strengstens verboten ist. Diese Zwischenwelt wo man sich die Rechte nehmen muss ist der Platz, an dem auch in dysfunktionalen Kulturen Projekte vorwärtsgebracht werden können. Fixiert man sich aber zu sehr auf das was man darf (Grün) verpasst man schnell einmal das was man könnte (Blau). Daher Augen auf!

Sergey says

It's definitely worth reading book since it once more time summarizes different patterns and habits in behavior of whole project teams or persons. If you are not a guru in project management, but already started noticing some common in teams behavior and cannot describe them in words only feel - this book will shed some light on these obvious patterns.
So common patterns in my experience are: Manana, Leakage. My "favorites": News Improvement, Hidden Beauty and the top rated in the real life of is Pile On.

Henrik Warne says

Adrenaline Junkies and Template Zombies is a collection of 86 patterns of project behaviour collected and documented by a group of 6 authors from the Atlantic Systems Guild.

Each pattern is presented with a title, a picture, a one- or two-sentence summary, and a few pages describing the pattern in more depth. This format works pretty well, and the book is both funny and very easy to read. However, when I finished reading the book and asked myself what I had learnt from it, I had to answer "Not much".

That's not to say it's a bad book, just that if you have been working in software development projects for a few years, there aren't that many new insights here. However, the book does a good job of singling out and labelling various project behaviours (usually bad ones), which is useful.

Of all the patterns in the book, the ones I liked the best were "The Blue Zone", "Practicing Endgame", "Mañana" and "Time Removes Cards from your Hand".

"The Blue Zone" describes the green zone, which is anything that is explicitly ordered or allowed by the project, and the red zone, which is anything explicitly forbidden. The blue zone is everything else, activities that are neither explicitly allowed, nor explicitly forbidden by the scope of the assignment. In the authors' opinion (and in mine, too), it is good to sometimes operate in the blue zone, in addition to in the green zone, in order to achieve the best outcome. Or, in the words of the quote ending the pattern: "The correct amount of anarchy on a project is not zero".

In "Practicing Endgame", the idea is that you should be thinking about and testing against your release criteria continuously, as opposed to leaving that till the end. The analogy given in this pattern is that of the university course, where you may have several tests throughout the term, in addition to the final exam. This "continuous" exam preparation gives better results than the one-off method of only having the final exam.

The last two of the patterns I liked the most both deal with time.

"Mañana" simply states that if your goal date is more than 30 to 90 days out, you need to set sub-goals that are within 30 to 90 days, in order to make the people on the project feel the right sense of urgency.

"Time Removes Cards from your Hand" describes how you have fewer and fewer options the longer you pretend that everything is fine, even though things are not fine. You might end up with many half-finished features, instead of a few completely finished features, and it might not be the most urgently needed features.

Except for the concept of the blue zone, which I like and which I had never seen explicitly described before, even the patterns I liked are not really teaching me a lot that I didn't already know.

In fact, if you are using agile methods like XP or Scrum, then you will recognize a lot of the patterns and advice as standard agile working procedures ("Straw Man" is another example of this).

On the other hand, there are a number of examples of anti-patterns from (it seems) process-heavy larger companies, for example "False Quality Gates" (documents are check for format, not contents), "Paper Mill" and "Orphaned Deliverables" (both deal with places where the measure of progress is documents, not working software), and "Cider House Rules" (rules are made by people unconnected to the project).

When it comes to the names given to the different patterns, there are some hits and some misses. A name that is both catchy and describes the pattern in a good way makes the pattern so much easier to remember. My favourite is "Template Zombies", which I think is pretty self-explanatory, but "One Throat to Choke" is also very good. But naming is hard, and there are many patterns that I feel have pretty awkward or non-descript names, like "Lease your soul" (about how to adopt new technology - I'm thinking more in terms of a tool-box than selling/leasing your soul to some new technology) and "System Development Lemming Cycle" (that the process used isn't tailored - but where did the lemmings come from?).

Another complaint is that the different patterns presented in the book are not organized around themes - instead they are just put in random order. I would have preferred if they were grouped together, since many of the patterns deal with related concepts.

So, in summary, the patterns in the book cover many different project behaviours. The descriptions are useful and well written, but if you have been involved in software development projects for a while, most of the patterns should already be familiar to you. Still, they may serve as a useful reminder - plus, you get (in many cases) snappy names for some of the behaviours, which may make them easier to diagnose and talk about.

Also, if you're interested in this book, check out episode 131 at Software Engineering Radio. That podcast is an interview with Tom DeMarco and Peter Hruschka about this book, and it is well worth listening to.