

The enterprise as story

The role of narrative in enterprise-architecture



Tom Graves



The enterprise as story

The role of narrative in
enterprise-architecture

©2012 Tom Graves

Last published on 2012-03-07



This is a Leanpub book, for sale at:

<http://leanpub.com/tb-estory>

Leanpub helps authors to self-publish in-progress ebooks.
We call this idea Lean Publishing. To learn more about
Lean Publishing, go to: <http://leanpub.com/manifesto>

To learn more about Leanpub, go to: <http://leanpub.com>

Contents

The enterprise as story	i
The role of narrative in enterprise-architecture .	i
Sample version	ii
Complete contents of the book	ii
Introduction	1
The enterprise as story	1
Who is this for?	1
What's in this book?	2
01: Everyday enterprise architecture	4
Application	7
Interlude	8
Application	10
02: What's the story?	12
Application	16
Interlude	18
Application	21

CONTENTS	ii
03: Narrative and story	22
Application	29
04: The structure of story	30
Application	42
Interlude	44
Application	45
05: Circular stories	46
Application	55
20: Wrap up the story	57
Application	59
Appendix: Sources and resources	60
Sources	60
The Tetradian Enterprise Architecture series . .	66
Other resources	67

The enterprise as story

The role of narrative in enterprise-architecture

Tom Graves

Tetradian Consulting

Published by

Tetradian Books

Unit 215, Communications House

9 St Johns Street, Colchester, Essex CO2 7NN

England

<http://www.tetradianbooks.com>

First published February 2012

ISBN 978-1-906681-34-0 (paperback)

ISBN 978-1-906681-35-7 (e-book)

© Tom Graves 2012

The right of Tom Graves to be identified as author of this work has been asserted in accordance with the Copyright Designs and Patents Act 1988.

Sample version

Complete contents of the book

Chapters shown in *italics* are included in this sample.

- *Introduction*
- *01: Everyday enterprise-architecture*
- *Interlude*
- *02: What's the story?*
- *Interlude*
- *03: Narrative and story*
- *04: The structure of story*
- *Interlude*
- *05: Circular stories*
- *Interlude*
- *06: Defining the story*
- *Interlude*
- *07: Where is the story?*
- *08: When is the story?*

- 09: Whose story?
- 10: Who owns the story?
- 11: Which story?
- Interlude
- 12: Finding the voice
- 13: Maintain the tension
- Interlude
- 14: Betraying the story
- 15: Start again
- 16: Start anywhere
- Interlude
- 17: Put myself in the story
- 18: The story-doctors
- Interlude
- 19: Win without the fight
- 20: *Wrap up the story*
- *Appendix: Sources and resources*

Acknowledgements

Amongst others, the following people kindly provided comments and feedback on themes in the various drafts of this book: Peter Bakker (NL), Stuart Boardman (NL), Shawn Callahan (AU), Pat Ferdinandi (US), Bas van Gils (NL), Nigel Green (GB), Paul Hobcraft (CH), Anders Jensen (AU), Gerold Kathan (AT), Cynthia Kurtz (US), Michael Margolis (US), James McGovern (US), Liz Poraj-Wilczynska (GB), Kevin Smith (GB), Michael Smith (MX), David Tebbutt (GB), David Topping (GB), Peter Tseglakof (AU)

Please note that, to preserve commercial and personal confidentiality, the stories and examples in this book will usually have been adapted, combined and in part fictionalised from experiences in a variety of contexts, and do not and are not intended to represent any specific individual or organisation.

Registered trademarks such as Zachman, TOGAF, FEAF, ITIL, Business Model Canvas etc are acknowledged as the intellectual property of the respective owners.

Introduction

The enterprise as story

“What’s the story?” - would that be an unusual phrase to start a business-book? Perhaps so, but it does seem appropriate here...

Once upon a time, we might say, I started out on a metaphorical journey to explore the role of story in the enterprise, and especially story in relation to enterprise-architectures. It’s been a strange journey, at times a very fraught one, yet also a worthwhile one. And one that does seem worth sharing with others. Hence, here, with you. A story about the enterprise *as* story.

Who is this for?

As with the other books in this series, I’ll assume that you’re working as an enterprise-architect, business-architect, strategist, process-designer or some such. That’s the ‘intended audience’, if you like.

But it actually doesn’t matter: unlike some of my more technical books, this one really *is* relevant for just about everyone who works in organisations. Story is for everyone: that’s the whole point of what you’ll find here.

What's in this book?

I'll say up front that this is a bit different from the usual approach I've used in other books. There is, of course, a good reason why.

As with most current conventional approaches to enterprise-architectures, most of my work to date has focussed on structure. Often a much broader view of structure than others might hold, perhaps - such as 'everything as a service', or the structures of relationships and intent - but structure nonetheless.

Yet that's only one side of the enterprise-architecture story. Literally so - because just as structure is everywhere and everything in enterprise-architecture, so is story. Story is *everywhere*. That's what makes it important. That's also, at times, what makes it hard.

The usual way to describe structure is through structure: hence the formal-style structure with most of my previous books, partitioned into themes and topics and the like, each with their own section about how to apply it in practice. In the same way, it seems the best approach to describe story is via story: hence a sort-of-story about story in enterprise-architecture, and about the stories, and the many layers and types and forms of story, that interweave with each other to make up the enterprise *as* story.

There *is* a structure here, of course, though we might describe it as somewhat post-modern in style: kind of structure-beyond-structure. Themes and phrases will recur

and repeat at times - as you may have noticed already - yet that too *is* intentional, the repetition providing not just re-emphasis but also often another view into the same overall 'holograph' that is the enterprise.

In the same way, the story may often seem fragmentary at times, things described only in part, as if seen from a distance - because that's exactly how we see a whole at each moment, with sense and structure of that whole emerging only over time, and only in its own way, beyond our direct command and control. (That's part of why this has been the most difficult book so far for me to write - but also one of the most rewarding.)

So it's a story about story, in a context where most people don't seem much to notice the stories, and for the most part don't seem to acknowledge their importance either. The story so far, perhaps?

Yet every book in this series is about practice, about *usefulness*, about ideas that we can put to immediate use in the everyday work of enterprise-architecture. So in keeping with the other books, there's one concession that I'll make here to the need for structure: I'll end each chapter with a brief summary-section, labelled 'Application', summarising key points from the preceding text, with suggestions about how to apply those ideas in your own real-world context. There's also a list of references and resources in an Appendix at the end of the book.

Anyway, enough for now: on to the story.

01: Everyday enterprise architecture

What's the story? It's another day here in everyday enterprise-architecture: all the usual tasks, the usual models, the usual analysis-work and review-meetings and all the rest. A lot of work still, to link everything together across that whole range of structures that we work with: data-structures, organisational-structures, process-structures, infrastructures, whatever.

All satisfying in its own way, of course. If nothing else, there's a great professional challenge for our team in all of this, in creating clarity, consistency, sense. And of real value to the business: we can prove that now, too.

And yet... there's something... odd... not quite right? Like there's something *missing*? A blind-spot, maybe; something huge, even; yet can't see what it might be *because* it isn't there? That kind of feeling. Disconcerting. Worrying. Or worse.

It's the same sort of feeling that was around when we first started our work on enterprise-architecture. Back then, everything was centred only around IT, an *IT-centric* view of the organisation, as if IT was the only thing that mattered in the whole enterprise; most of what they'd called 'business-architecture' would be more accurately described as 'anything not-IT that might affect IT'. Useless, really; or misleading, at any rate. But things *have* changed

over the past few years: there's a lot more awareness of business-strategy, of business *as* business. The claim to be *enterprise*-architecture now looks a lot more credible these days: much more of a meeting of minds with the business-folk.

Yet still there are those nagging doubts... We see it, perhaps, in that so often there's so much of a *struggle* to get things going, to get people onside. It's not a problem of scope any more: we really do have much more of an enterprise-wide scope. It seems more that although our models and so on make perfect sense to us, they don't seem to make sense to anyone else. They're not *connecting*, somehow. But why not? That doesn't make sense either...

That nagging uncertainty takes me back to an odd incident at the launch of a new version of a well-known enterprise-architecture framework. In the midst of all the hoo-ha and self-celebration, one of their team pulls me aside. "You know what's missing in all of this?" he says, in a whisper, as if afraid to be overheard. "There's no place in it anywhere for *people*."

He's right: there isn't. People are mentioned occasionally almost in passing, perhaps as inputs to processes, but nowhere *as* people in their own right. Odd... Yet what does that absence imply for the architecture - or for the enterprise that uses that architecture?

Another memory comes up. This time I'm on a train, reading Matthew Frederick's *101 Things I Learned in Architecture School*. It's about building-architecture, not

enterprise-architecture, yet it's clear that many of the same principles do apply to both. The book is laid out in simple two-page spreads, sketch-drawing on one side, a brief pithy summary on the other. As I turn the page, the text all but leaps out at me:

Two points of view on architecture:

*Architecture is an exercise in **truth**.* A proper building is responsible to universal knowledge and is wholly honest in the expression of its functions and materials.

*Architecture is an exercise in **narrative**.* Architecture is a vehicle for the telling of stories, a canvas for relaying societal myths, a stage for the theater of everyday life.

Truth and narrative; the structure and the story. All of those conventional EA-frameworks focus on the certainties of structure - "it's all about IT processes and data", as one somewhat-myopic colleague used to insist - leaving no place within them for story *as* story. Yet if there's no story, there's no place for people - which is *why* there's no place in it for people.

Which is why, in that kind of purported 'enterprise'-architecture, there's actually no place in it for *enterprise*. The enterprise *is* people: enterprise is what people *do*. But if the architecture has no place for people, then those people can't be enterpris_ing_ within it: and we then wonder why we have enterprises that fail...

And no wonder we have a problem with the architecture, too: in an all too literal sense, if we only work on structure, we'd only be delivering one half of the actual architecture. Oops...

But if enterprise-architecture is missing its story, what *is* that story? How do we find it? And how do we embed it or link it into the architectures that we already have?

What's the story? Clearly, something that needs to be explored...

Application

- Is that true for you? - that the story of enterprise architecture seems to be missing its story?
- What, to you, is 'story'? Where - if at all - does it apply in enterprise-architecture?
- Which types of stories are included at present in the architecture? How are they used? What do they add, and how, and why?
- What are the missing stories? - the types of stories that we need, but aren't there, or aren't visible? And if the stories are missing, what impact does that absence have on your enterprise-architecture, and on the enterprise as a whole?

Interlude

Earlier this week I'd spent a long session with two colleagues, talking about how our respective business-models have had to adapt over the years, to re-align with changing needs and changing perceptions in the marketplace. Without much noticing that we're doing so, we describe each change with a story, about letting go of the legacy of the past, and of what we learn from and through the change. A biography of a business, told in stories. And stories within stories, as anchor for meaning, and as compass: by changing the story, we change the direction.

So it's the weekend now, a different café, a different conversation. A friend's 50th-birthday party: he's proudly wearing a badge that says "21 Today! (29th anniversary)" - a small icon that says so much about our culture's attitudes to age. I'm seated opposite his late-teenage daughter, and we strike up a conversation about what she's doing, her classes at college, her photography, her hopes and fears for the future. It doesn't take long until the stories start to flow - some of them taking me back too many decades, to when I faced much the same fraught times and temperaments. Stories as connection; stories as challenge...

The meal over, we continue the conversation outside. Out there in the street, there's a young man of about her age, busking with a full drum-kit and backing-track. He's drawn quite a crowd there, though many of them more interested in the antics of the old man who's dancing quietly to the music, over on the far side. He's well into

his eighties, at a guess, and a lot of people are laughing at him: I'm not, because it's clear he's not 'showing off', or knowingly making a fool of himself in public, he knows exactly what he's doing, with the music, and the dance. He knows he's being mocked, but he doesn't care: he's only interested in the music, and what it means to him. And he dances on, with care, precision, locked exactly to the beat, and with much more energy and stamina than one might expect from someone of that age. A drummer himself, I'd guess, not so many years ago.

I'd been searching for a story to show her how we can't depend on inspiration alone, that it comes and goes whether we like it or not, and hence how and why we need all those 'boring' predictable patterns to fall back on, to keep us going and to continue to do *something* useful until the inspiration does return. Which it always does - *if* we allow it to do so, and if we don't try to control how and when it does. And yet here's the perfect story, right here in the street, in the old man's dance. As the music rolls on, most of the time he's right there, 'in the groove'. From time to time, though, he loses the connection: for a moment, the inspiration that drives the dance is gone. Yet he doesn't try to control it, to force the dance; and neither does he just give up and stop. Instead, he falls back to a pattern, swaying gently from side to side, still listening closely with his whole body to the beat, keeping a space open for the inspiration to return. Which it does: suddenly he's back in the groove again, dancing with 'the flow' as much as with the dance itself. A real inspiration: and, clearly, a lot of

stories behind the surface story.

Application

- Looking at your own organisation - or your own business-life - in what ways has 'the story' changed over the years? At each stage, what was let go, what was learned? What was lost? What was more a liberation? What's the story there?
- Looking back, what stories do you tell yourself, and others, about those times - both of the periods of continuity, when things seemed stable for a while, and of the transitions, those times of uncertainty that were somehow 'between stories'?
- What cultural clues do you see from the small stories, such as in that 'not-21st-birthday' badge? What can you learn from similar 'small stories' within your own organisation?
- In what ways do stories create a connection between you and others? How do they connect across time, across generations? In what ways could such stories help to create connection and shared-learning across your own organisation?
- What are the stories that create energy, stamina, drive - even in the face of mockery and other opposition? Where would such stories have most impact in your own enterprise?

- Where do you find stories and parables to illustrate particular points in your architecture? How do you create the space to allow yourself to notice suitable stories serendipitously, from what's happening around you in the moment?

02: What's the story?

Staring at the whiteboard, for hours on end, trying to find *any* pattern in that blizzard of sticky-notes and scrawls. One question keeps coming back, time after time: *What's the story?*

Stuck. We all know *that* feeling...

Step back a bit: what's the story behind the story? Well, that part's straightforward, I guess. I'd been invited to do a presentation at a conference on enterprise-architecture. I knew the audience would be broader than just the stereotypic 'finance, banking, insurance, tax' of so much mainstream EA, so it would give me a chance to try something new: the role of story in enterprise-architecture. I'd already explored various aspects of this in various posts on my weblog, so it shouldn't be difficult. And yet it is: I'm stuck, going round and round, on just one question: What's the story? What's the theme that will hold it all together? I can find enough of a story-line with which to do the presentation - but not story *itself*. There's something strange here that I can't pin down...

Structure and story; the structure is already there throughout our enterprise-architectures, but the story isn't. Yet looking across the EA discipline as a whole, everywhere there's the same kind of imbalance as that between structure and story:

- *content versus context* - just about everything we

see describes content, often seemingly divorced from any context...

- *'control' versus trust* - a lot of emphasis on would-be 'control', not much on trust...
- *organisation versus enterprise* - many people seem to think that the organisation *is* the enterprise...
- *certainty versus uncertainty* - it's almost like uncertainty isn't allowed - yet the real world *is* uncertain...
- *sameness versus difference* - again, the same drive to try to make everything the same, certain, predictable - which the real world isn't...
- *rules versus principles* - huge emphasis on 'the letter of the law', yet not much about the *intent* in 'the spirit of the law'...
- *machines versus people* - not much room for people *as* people, pretty much anywhere...

The balance between structure and story - or lack of it - seems similarly skewed, yet also seemingly linked to everything else. So perhaps if we focus on story, we might also help bring those other themes back into better balance too.

So what *is* story? And where *does* it occur within our enterprise-architectures?

The short answer is: *everywhere*.

Every use-case is a story - a story of what we want to happen.

Every scenario is a story of what *might* happen.

Every customer-journey is a story-in-progress; likewise every transaction, even right down at the level of two computer-systems talking to each other; a transaction-protocol is a proto-story.

In marketing, the whole point of a brand is that it tells a story - a story of hope, desire, intent, of use and usefulness, of longing and belonging.

Every learning is a change of story; every change-project likewise aims to change a story.

A supply-chain is a story; a strategy is a story; even the enterprise itself is a story.

And so on, and so on, and so on: story is so *much* 'everywhere' that it's sometimes hard to see...

So *why* story? What's so important about story? Why does it matter?

This time the answer's not quite so short, but in essence it's the distinction between *content* and *context* - the 'things' of the world, versus the *usefulness* of those things. The physical world is made of atoms, but the human world is made of stories. Within a business, within its architecture, we might have all the 'things' we need to make everything work; but without the stories, there's no meaning, no purpose, no reason or drive to *do* anything with those

things - and hence no *enterprise*. Which also means no business. Hence why, yes, this *matters*...

More subtle, perhaps, is a trap pointed out many centuries ago by Lao Tsu, in the *Tao Te Ching*:

“Profit comes from what is there;
usefulness from what is not there”.

If we're developing an architecture for a commercial business, there'll no doubt be a lot of focus on profit - sometimes almost to the exclusion of everything else. Yet whilst the profit may arise from 'what is there' - from "the ten thousand things", to use Lao Tsu's phrase - the 'usefulness', the way we *create* that profit, comes from 'what is not there', the connections *between* things, the stories. So if we can't see the stories - because they're seemingly 'not there' - we'd then have no way to reach that profit. Hence, again, this *matters*.

Finally, another quote that caught my eye the other day, this time from Christina Baldwin:

“Words are how we think; stories are how we
link”

Words are easy; it's finding the story that's hard. And the same is true in our architectures, the age-old distinction between 'boxes and lines'. The boxes in all those myriad diagrams represent the 'things', whilst the lines represent

the connections *between* things. And yet so many of those diagrams are static - literally so, since they claim to describe some past or current or future state. What we rarely see are the dynamics, the *stories* that traverse those lines connecting everything together. So to make the architecture work - to bring it to life - we need to balance the structure with the stories.

Finding the stories might well be a way to get unstuck, too...

Application

- Where do you get stuck - in particular, stuck in terms of ideas and issues at work? What's the story behind that sense of 'stuckness'?
- What are the stories that you see in your architecture, your organisation, your enterprise? What is it that *makes* it a story?
- "Profit comes from what is there; usefulness from what is not there" - What does this suggest about your own organisation and its architecture? Is the focus only on 'what is there', or does it also include 'what is not there'?
- And by what means can you see 'what is not there'? Or explain it, or even describe it, in your architecture? How can you describe what's missing from 'what is not there'?

- 'Boxes and lines': you'll see plenty of those in your architecture diagrams. But what are the stories that link those 'boxes' together? And how *do* these stories traverse those metaphoric lines?

Interlude

It's *all* about story. All of it.

Enterprise-architecture is all about story. The enterprise itself is a story; but the practice of enterprise-architecture is all about stories too. Let me tell you a story...

There once was this half-crazed guy who used to go on about an even crazier idea that there might be a bit more to enterprise-architecture than just, well, IT-boxes and suchlike. That there was a bit more to the story than that.

It starts, like most good stories, a long time ago. Turns out that whilst he'd arrived at EA via the usual IT-journey, from years of assembly-language through to database-design through to data-architecture and information and the rest, that wasn't where he really came from. (I don't think he's the long-lost Prince of Multigravia, though - it's not that kind of story. Sorry.)

He'd actually started out in graphic design, getting side-tracked into software and stuff to try to get typesetting-systems to work better. And he hadn't forgotten the designer's way of thinking about things (which these days goes by the fancy term of 'design-thinking', but it wasn't called anything much back then). It was always about thinking about the big-picture at the same time as looking at the small-picture, and keeping the tension in balance whilst going in deeper and smaller and smaller and smaller, whilst still keeping the big-picture and the bigger-picture and the really-really-big-picture all in view at the same

time. Kinda crazy-making, but that's designers for you, of course.

And then something happened. (Yep, that's a phrase that comes a lot in stories.) He was working on enterprise-architecture by then, the usual tedium of tracking down all those tiddy little Access databases and spreadsheets that were being used way outside of their scope or capability. Oh joys. But one area he was asked to look at in this was quality-management. Which brought up a simple yet surprisingly scary question: what *is* quality, anyway?

Quality can be very tricky indeed: not a popular topic with many business-folks. So unpopular in this case that the quality-manager had committed suicide. (That part isn't a story, sadly...) So how are they going to manage quality? We know how to do it, they said - we can do it all with software! Buy an off-the-shelf package from one of the big vendors, plug it in, roll it out to the whole workforce - there, problem solved! Easy! It's a few million bucks and ongoing but so what, it's just 'fit and forget', isn't it...?

Uh. No. Even our half-crazed anti-hero could see that it wouldn't work. The problem was that lots of people wanted to believe that it would. Lots of serious business people with serious career-ambitions, and with serious access to lots of other people's money. (Are these the villains of the story? Perhaps - but we'd better not say so in public if we want to keep our jobs?) A tricky enterprise-architecture problem, that one... But with a lot of hunting around, amongst the few mostly hidden backroom-boys

still holding out the flag for the not-quite-lost quality-cause, we found an in-house team who'd developed a quality-system that really did work. All done on little pieces of paper. No IT at all.

Sure, they were moving some parts onto software, but that was just a simple customisation of a 'knowledge-management' system for which we already had a site-wide licence, and which almost no-one else seemed to be using anyway. Straight away a serious saving of several million dollars. But the main point was that it worked. *And it was all about stories.* Making sense, through stories.

Getting people to work together, to get the work together and make it work better.

Finding the best way to do the work, in whatever combination of people, machines and IT would be the best fit to that particular context.

Exploring, enquiring, endless seeking, ceaselessly improving. A commitment to quality; an enterprise in itself.

All through stories.

Which is itself a story.

So where are all of those stories in our usual narratives about enterprise-architecture? Uh... still nowhere to be seen? Oops...

Perhaps this all too usual story of enterprise-architecture needs a different ending...?

Application

- What are *your* architecture stories? When you're faced with a horror-story such as that of the quality-manager above, where would you find the counter-story to make things right?
- Do you work with people who assume that every business-problem can be solved with some kind of pre-packaged IT-system? - and that that option must by definition be the best choice, solely *because* it's IT-based, and therefore 'known' and 'certain'?
- If so, how do you get your colleagues to explore the wider story, and perhaps seek for simpler, more maintainable solutions? How do you engage them in a story that might revolve around *people*, rather than solely around IT?

03: Narrative and story

So what *is* story, anyway? What is it that makes story different from anything else? And when we're describing the structures of an architecture, isn't that a story too?

Well, sort-of, I guess... Structure itself is just structure: there's no story there, as such. But we tell stories *about* structure, *around* structure; structure as backdrop to story, the stage upon which stories are set. Structure is structure: it doesn't have much *meaning* until there's a story. Architecture is both structure *and* story, is both truth *and* narrative; the one doesn't make much sense without the other.

Yet if we look at most of the EA tools that we have, and the EA methods that we have, they're all about structure. They're very good on structure - no doubt about that. Unfortunately, though, they're *not* good on narrative, or story. There are a few notable exceptions, but for the others...? - well, apart from a few brief excursions into use-cases and the like, they're not much use on narrative... not so much 'no-story' as a negation of story itself.

Oh.

Which is a much more serious problem than it looks, because in practice, most of our enterprise-architecture work is actually about stories. Stories upon stories: lots of them.

Again, look around: story is *everywhere* - hard to see only

because it *is* so much ‘everywhere’. Every strategy tells a story - a story of a different future. Every merger or demerger or restructure or reorganisation or re-whatever is a story, a change of story. On the smaller scale, a business-scenario is a story. A use-case is a story. From as-is to to-be is a story. A typical application-consolidation effort is a story too, about how to clean up the tangle of this-doesn’t-go-with-that, and change it to a new story of and-they-all-lived-happily-ever-after (until the next consolidation, anyway). It’s all stories.

Every requirement implies a story. The work of an Agile team is all about co-creating a shared story. And getting people to work together is a story in itself, and one that in itself is so often made up of people sharing their different perspectives on what should end up as a shared story - perhaps across the whole enterprise.

Enterprise-architecture, in this sense, is all about supporting those stories. Every model tells a story, a record of decisions, options, choices. We can use each model to elicit further stories: people disagree with the choice, perhaps give us their story of why the choice needs to change; perhaps they agree, and the story helps to reaffirm and reinforce that choice, that story.

Yet at the moment, almost none of that is in the models. Or, for that matter, the methods. We’re somehow supposed to know it’s all about story - but somehow pretend it’s all about the IT instead. Odd...

And when we stop to think about it, enterprise-architects

don't actually *do* much other than tell stories, or get others to tell stories. We don't do much development-work, perhaps not any: that's the solution-architects' job, and they'll often get annoyed if we tread too much on their turf, their story. We don't have much authority - especially beyond the borders of our own organisation. In most cases all we *can* do is influence, cajole, guide. And the way we do that is by creating a story.

It's all about stories.

And though I hadn't noticed it until now, that lack of support for the story is what's been so frustrating for me about so many of the existing toolsets, too: it's not that their near-exclusive focus on structure is somehow 'wrong' or whatever, but it's because it stifles the story. Some of the toolsets are so constrained and so clunky that it's like being a captive in kindergarten again, where the only permissible poems must be modelled on 'Mary Had A Little Lamb'. The obsessive IT-centrism in so much architecture is like the self-centred bar-room bore, who insists that they have to be the hero of everyone's else story. (The business-centrism of so many business-architecture tools is no better, by the way.) And a half-assed, half-complete story is no story at all. No fun for anyone else, anyway.

As enterprise-architects, we need to engage people in change, in a vision of something that works better than they have at present. And that's why we need toolsets and methods that can cover the whole scope, the whole story of the enterprise, and support us in our storytelling of that story -

because we need them to engage themselves in that broader story.

Enterprise-architecture: it's all about stories. So it might help to remember that fact - by telling a story or two, perhaps?

Yet how *do* we tell an architecture-story? And story and narrative: aren't they the same? Again, the answer's 'sort-of': they're both about sequence of action in context, "something happened, and then something else happened, and then something else...". With narrative, that's just about all we get; but a story is, well, a *story*.

Narrative is important - don't get me wrong about that. We use narrative all the time in architectures. For example, consider a UML (Unified Modelling Language) Sequence Diagram or State Transition Diagram - all the *behaviour-diagrams*, as contrasted with UML's structure-diagrams. Each of those is a visual description of 'something should happen, and then something else should happen', with branches in the sequence to show what should happen in this case, or that case, or the other case. Something happens; and then something else happens. That's narrative.

There are variants of that kind of narrative, of course. There's a *protocol*, a kind of pre-planned narrative between two entities, describing who should do what and when, and in response to what and when - again, a sequence of action in context, but this time of *shared*-action in context. And the overall 'conversation' can involve more than two entities, of course: consider a supply-chain sequence from

supplier's-supplier to customer's-customer, or the links between swimlanes in a BPMN (Business Process Modeling Notation) process-diagram. Yet it's still just a narrative: something happens, and then something else happens.

The problem with narrative alone is that it's, well, *boring*... It's like sitting in on someone else's slideshow of "What we saw on our holidays": sure, something happened, and then something else happened, all nicely illustrated with pretty pictures and the like - but so what? Where's the *story*?

So much of our current architectures are like that slideshow: lots of narrative, lots of structure in the background, lots of pretty diagrams, but where's the story? There's nothing to *engage* people's attention, to answer the inevitable questions:

- what does it mean?
- what's the point?
- where do *I* fit in this story?
- what's in it for me?

That's the real difference between 'mere narrative', and story. Narrative tells us the what and how and when and who and sometimes where; and, when well-done, does it well. But story adds that crucial all-too-often-absent element of *why* - the *purpose*, the *meaning* behind the narrative.

Story gives us the reason why that sequence of action in context should happen in the first place. Story shows us what we can learn from what happened. Narrative alone is often too abstract to make sense; story makes it *real*, tangible, concrete - despite being just as imaginary as the narrative itself.

That's why *story*, rather than solely the narrative. That's why story *matters* in our architecture, why structure alone is not enough - because without the story that accompanies that description of the structure, there's no meaning, no point.

With the story, there *is* a point. A stakeholder asks us "What's in it for me?": a story explains exactly what's in it for them, places them *in* the story, makes it *their* story too. Many architects I speak with complain about how difficult it is to 'sell' their architecture to the stakeholders: yet if we build it around story, it sells itself. A lot simpler than hard-sell...

One key difference here between narrative and story is 'the unexpected': "we were doing this, and doing that, and then doing this as well, *and then something unexpected happened*". It engages the people's attention; it introduces the possibility, of, well, *possibility*, really. Scary, yet hopeful, all at the same time.

For me, working on this book, the main 'the unexpected' in the story so far isn't about what's happened, but much more about what *hasn't* happened. By this stage I would expect to be working to an outline that I'd planned out, step

by step, weeks or even months ago. But that just hasn't happened. I can't find the structure; can't find the story. All I have at present is a huge pile of notes, a great big hole of uncertainty that's loosely labelled 'Magic Happens Here' - so often the case in architecture-development! - and a single tag-line to which I keep coming back, and coming back, and coming back: *What's the story?*

The story's in there somewhere: I'm certain of that. Yet at present that's just about the only thing I *am* certain about: for everything else, right now, I'm stuck with having to 'wing it' until more of the clarity comes through. Uncomfortable. Very.

And yet that's not at all unusual at this stage of most architecture-work. So often I've sat with colleagues, staring at wall-fulls of diagrams and charts and summaries and other information, waiting for some kind of sense to settle out of the chaos in front of us. Sometimes it just takes its own time: and the more we push it, the further away it seems to get.

So in a sense, right now the story here is that there isn't a story. Except that there *is* a story. Somewhere. Just have to wait it out, I guess. Or rather, keep going, keep it moving, working on some part of structure that might help the story to coalesce. Structure and story intertwine: working on story helps us find the right structure, and working on structure helps us find the right story.

Which suggests that the best thing to work on for now would be the structure of story itself. Play with that for

a while, and see what comes out of that.

Application

- What examples of narrative - sequences of action in context - do you see in your architectures?
- Where's the story? Much of architecture will focus on What and How, but where's the Why that makes it a *story*, something that *engages*? Where's the uncertainty that engages the interest?
- How much of your architecture-development consists of 'winging it', finding ways to cope with the unexpected, before you can get back the relative safety of structure, of the known? What's the story there?
- When you get stuck - especially at the big-picture level, as here - what do you do to get unstuck? What structure, what process, what habits, what tactics, do you turn to, in order to get the ball rolling again?

04: The structure of story

What *is* a story? What's the *structure* of story?

Perhaps the best people we could turn to for help at this point would be the professional storytellers. For example, how does Hollywood structure a story? What do *they* think story is?

In a sense, they have the opposite problem to ours: our visible end-product is structure, and we need to support it with story; whereas their end-product is story, and they need to support it with structure. Overall, though, it *is* the same kind of problem: how to balance structure and story. The Goldilocks Challenge, we might say: not too much of either one or the other, but just the right amount of each.

And writing for the stage or screen - whether the big screen, the small screen, or the tiny handheld one - is a big business. *Very* big. Hence no surprise there's a *lot* of study on story and structure, and how to make them work together. The real challenge is to find the parts of structure that work well with *our* kind of story - the story of the enterprise and its architectures.

Some parts of story-structure - such as character-arc, for example - either don't fit with what we need, or at first may not seem to make much sense in our context. Yet there's a lot that does align very well with our needs in architecture: viewpoint, for example; or genre and mood,

serial versus series, establishing the story, pacing, setup and payoff, possibility and probability, the real importance of the unexpected for surprise or delight, the problem of plot-holes - it's all there, ready for us to use in the architecture-story too. We just need to know how to adapt it, apply it...

One place we could start is with the screenplay, which, in a sense, is the architecture for the on-screen story. We could note, for example, just how efficient - or more, how *effective* - a screenplay has to be: it's no different from any other architecture in that respect. It takes a lot of skill, and a lot of work, to trim the five-hundred-plus pages of a Harry Potter novel all the way down to a sparse hundred pages of dialogue and direction.

Most standard screenplays are structured such that each page represents just one minute of screen-time: and everything has to be there on that single page. Every page has to carry the *essence* of the story, at every moment, without ever losing connection with that core; and every page, every line, every phrase, has to carry the story forward, with not a single word to waste. Taut; elegant; spare: those are some of the words we might use to describe a great screenplay.

And yet that screenplay is just the start of the production story. Look at those end-credits again: every film or show is a huge collaborative effort, involving tens, hundreds, maybe thousands of people, all with different skills to bear. Each will view the story and screenplay in their different

way; each will bring their own insights to the story. Which means they'll usually want to change the story, too, to make things easier or more interesting for their own department. Hence the role of the architects - the director and, perhaps even more, the producer - to invite and encourage all these different views, and yet still hold true at all times to the essence of the story. That's not easy: we've all seen films that fall apart into a muddled mess somewhere. Yet the real masters make it seem so easy, so seamless, that we don't notice it at all: an interesting and important criterion for success...

Anyway, let's get back to the structure of story, in terms of the architecture of the enterprise.

If we ask the storytellers what to do here, they'd probably point us to the 'Hero's Journey' story-structure. This structure is so often used that some people claim that it's the *only* possible story - which I somewhat doubt, but it's certainly true that parts of it have become Hollywood clichés. For example, if you've ever wondered why there's an abrupt change around halfway through a story, or why there's some kind of death or bleak ending at the three-quarter point, or why the hero or some other key character seems to die (but usually recovers) just before the end, yes, it's because they're using the Hero's Journey pattern.

Which, at first glance, suggests we need to know that pattern if we're going to have any chance to understand the story side of enterprise-architecture.

So, start with a reminder of that point that screenplays

are written in a format where one page equates to one minute of story-time. We could use page-counts to indicate how far through the story each key point in that pattern would occur; but since a typical Hollywood story runs for around a hundred minutes, we'll make it simpler and use percentages instead:

- *Before the start*: “someone toils long into the night”, often for many years.

This is the ‘backstory’ for the story, which we’d usually find out through small dribs-and-drabs of information or ‘exposition’ as the story moves along. For a business or other enterprise, there’s usually a lot of backstory: as someone put it, “it takes many years’ hard work to become an overnight success...”.

- *During 1-5%*: start with a strong opening image, to establish the ‘ordinary world’ for our lead character.

For most enterprise-architecture, this would be that everyday world of structures and models, all centred around IT and the like. And yet there’d also be those niggling doubts and subtle hints that suggest something is seriously wrong - which leads us to:

- *At around 5%*: a trigger-event, an ‘inciting incident’ or ‘call to adventure’.

Sometimes this might be ‘something bad happens’, sometimes something unexpectedly good, but either way it’s a wake-up call of some kind. In business, this might just be the ‘initiating event’ for a business-process; for me here, the inciting-incident was two-fold, that comment about “there’s no nowhere for people”, and the note about ‘architecture as narrative’.

- *During 5-10%:* outline the nature of the ‘special-world’.

One of the key points here is that the inciting-incident shows us a new possibility of some kind - referred to in the Hero’s Journey pattern as the ‘special world’. We then have an apparent choice, either to go into that special-world, or stay where we are. If we stay where we are, of course, there’ll be no story - or, in sales terms, there’ll be no sale.

- *At around 10%:* setup is complete, and initial opposition identified or implied...

For this example of enterprise-architecture, we’ve established that story is everywhere in the enterprise, so story is clearly important - yet just about everyone else is thinking only in terms of structure. There’s certainly a conflict there - and conflict, we’re told, is the core to any good story.

- *During 10-25%:* ...the hero refuses the call to adventure.

And I'll admit that's probably what I'm doing right now: still trying to explain story in terms of structure, rather than 'story *as* story'... In the story-pattern, this is a stage that can go on for quite a while, with increasing pressures to move into the special-world, yet still persistent evasions of the call. (Salesfolk would recognise this stage as 'objections to the offer'.) But all that dithering and procrastination can't go on forever, because:

- *At around 25%:* point of no-return, an irreversible commitment - also known as 'crossing the first threshold'.

It's a decision-point - sometimes forced on the hero, sometimes a deliberate choice, but the key point is that now there's no turning back: we're in the special-world. Another key facet is that this isn't about analysis any more: like all real decisions at the moment of action, it's an *emotional* choice, not a 'rational' one. (Skilled sales-folk know that this is the moment to *stop* talking, and allow space for the choice to take place.)

- *During 25-45%:* 'fun and games' with tests, allies and enemies.

It's often incidents in this section that will end up in the trailer for the film. For a conventional story, it's here that we would establish 'the B-story', a secondary thread

cutting across the theme - such as the standard-issue 'love-interest' in classic Hollywood action-movies. (Sales-folk might recognise this as the initial follow-up to the sale, showing extra options and possibilities that anchor the purchaser's satisfaction at their choice.)

Another common theme here, which usually won't become clear until later, is that during this stage we either 'go for the wrong goal', or a goal that is narrower in scope than that which is needed to resolve the key theme of the story. That's something I still need to identify for *this* context, on story in enterprise-architecture...

- *During 45-50%: approach to the inmost cave.*

What happens in that previous 'fun and games' is what we might describe as exploration of the *rational* outcomes of that decision to enter the special-world. At some key point we also start to hit up against the *emotional* side as well - and it's the emotion that drives the whole story. (Sales-folk would recognise this as another key signal to stop talking, and *listen* for a change in direction.)

- *At around 50%: abrupt stop at midpoint - 'cross the second threshold', endure the ordeal and take the reward*

There's an important emotional challenge at this point, usually where the protagonist must face huge inner doubts,

and often either gains a new but more challenging ally, or is forced to go on alone without a key ally. (Sales-folk would know this as the first stage of ‘buyer’s remorse’, where the prospect hits up against the emotional downside of the purchase, and wants to back out. It’s essential to acknowledge that this *is* a challenge that the buyer must face alone: any help from the sales-person at this point will only make things worse, and possibly cost the sale.)

- *During 50-75%: ‘pursuit on the road’* - renewed challenges, the antagonists close in

The ‘fun and games’ of the previous stage return, but this time with renewed intensity, and often with a change of allies, hence a new set of interpersonal dynamics. (Sales-folk might recognise this as the point where they hand over to another colleague to answer detailed technical questions, for example - and they too have to trust that the technical guy won’t screw up the sale.)

Importantly, this section continues to ‘go for the wrong goal’ - still tackling only a subset of the real issues that drive the initial ‘call to adventure’. (A first-hand example for many enterprise-architects is the way in which the over-emphasis on IT had masked the fact that even the IT-issues can only be resolved by becoming more aware of impacts across the whole shared-enterprise.)

- *At around 75%: the ‘all is lost’ point, followed by the ‘Dark Night Of The Soul’; then reversal, to ‘cross the third threshold’*

Often this takes the form of a ‘double-whammy’: an intensely personal and painful realisation that this path just isn’t going to work - often accompanied by the loss of the most important ally - followed by a dawn of understanding that this wasn’t the right path in the first place. There’s a brief period of mourning, and then set off once more in a new direction. (Sales-folk would know this as the inverse version of ‘buyer’s remorse’, where the buyer suddenly realises that what they’ve chosen and committed to will not actually do the task they need. This is the moment at which carefully-placed options for upselling become possible.)

- *During 75-85%:* increasing intensity and renewed pursuit in the push towards the new goal

This is another version of ‘fun and games’, except that now there’s much more clarity about what the goal really is - and also the challenges to face in getting there.

- *During 85-90%:* a setback highlights the final challenge, and a rethink of the plan

The goal is clear, but the tactics to get there are not: this is the point about the often quite lengthy challenge-and-setback here. (Sales-folk would know this as the moment to re-frame the story, particularly if there’s a need to support an upsell.)

- *During 90-95%:* climax - often as ‘death and resurrection’ - followed by final resolution

After a final struggle - often primarily personal and emotional - the goal is at last achieved, and the protagonist 'reborn' in new form. (Sales-folk would recognise this as the actual moment of sale.)

- *During 95-100%: final resolution - 'return with the elixir'*

The first half of the final-resolution is for the protagonist to know that the quest is over; the second half here is often a more public acknowledgement of the fact - a recognition by others rather than solely by self. There's often an explicit choice as to return back to the 'ordinary world' of the start, or to remain in the 'special world' defined by the story; and also usually some form of 'boon' or prize or reward at this point. (From a sales perspective, this is the key 'customer-satisfaction' moment.)

Once that 'return with the elixir' is complete, the ending needs to come as quickly yet cleanly as possible: it's all over bar the wrap-up and, of course, a setup for a possible sequel (otherwise known in the sales-context as 'repeat custom' - a point we'll probably return to later).

So that's the Hero's Journey story-pattern. It might sound a bit alien at first, but if we look around with our eyes attuned, we'll see that pattern surprisingly often in business: it matches well with many sales-processes, for example, as can be seen above.

And each traverse through a business-process is a self-contained story with its own actors, actions and events:

we'll often find that some form of the Hero's Journey pattern fits well in those contexts too.

Where it *doesn't* fit so well is at the larger scale, for the enterprise as a whole, and especially so over the longer-term. The reason for this is to do with the overall nature of the story: single-shot, sequel, series or serial.

Most conventional films frame their stories as *single-shot* or *sequel*:

- A ***single-shot story*** is a once-off, typically in some form of Hero's Journey structure, with an emphasis on achieving a single goal or change.

I haven't seen many equivalents of this in the enterprise-context: a grand objective such as 'First Man On The Moon' or 'Eradicate Malaria' might seem to fit, but the respective organisation usually continues on in some form once the objective is achieved, and in fact is at risk of literally 'losing its story' if it aligns itself too strongly with the objective.

- A ***sequel*** is a once-off story, repeated, often without the emotive drivers that underpinned the original story.

In an enterprise-context, the sequels to 'First Man On The Moon' provide a well-known example: the energy and purpose of the NASA story there seemed all but lost, other

than the unintended genre-shift to ‘thriller’ with Apollo 13...

By contrast, most enterprise-stories are more likely to take the form of a TV-style *series* or *serial*:

- A ***series*** is a set of similar episodes in much the same contextual space, and often with some of the same lead-players, yet also often not much direct connection between the episodes themselves.
- A ***serial*** has a set of repeated patterns that provide a strong continuity, chaining all the episodes together into a single ‘grand story’.

A project-oriented organisation or consultancy will tend to look like a series-type story; a production-oriented organisation will tend to follow a serial-type enterprise-story.

A series can be quite fragmented, but a serial *must* take a long-term view: if it doesn’t maintain that constant continuity, we risk ending up with a disjointed mess that doesn’t make business sense - which is not a good idea.

So whilst the Hero’s Journey type of story-structure often helps us make sense of a sales-process or business-process or a linear ‘one-shot’ project, it doesn’t work so well for iterative stories, such as a series, or where many different stories weave through each other, such as in a serial. At the larger scale, we need a different type of story-structure - which is what we’ll turn to next.

Application

- If your enterprise was a film, which film would it be? What film-genre would it fit best - a detective-story, a romantic-comedy, a disaster-movie? Who would you place as the lead-actors? And why?
- “Taut; elegant; spare: those are some of the words that we might use to describe a great screenplay.” If your organisation’s current architecture was a screenplay, what adjectives would you use to describe it? How well does that screenplay hold the essence of your enterprise-story?
- Rather than what film it most resembles right now, which film *should* it be? What’s the difference? To make that happen, what would need to change in that story you see right now?
- Your enterprise too will likely be “a huge collaborative effort”, with many different skills and viewpoints brought to bear upon it. Who are all those people, these characters or stakeholders? What views and skills do they each bring to the story? What’s needed to bring out the best from each, and maintain the balance between them to create a story that is a seamless, meaningful, unified whole?
- Where do you see hints of the Hero’s Journey story-pattern play out within your own enterprise? Where do you *not* see that pattern? What are the key

differences between those contexts, where you see that pattern, and where you don't?

- Is your organisation's story more a single-shot, a sequel, a series, or a serial? Why would you say this? What are the characteristics you see in your enterprise that would suggest to frame that story in one way rather than another?

Interlude

Start work on this again this morning, yet again with a flood of new ideas, about storymind and storyworld, about storytelling and storylistening, about the relationships between character and culture, and a whole lot more.

None of which will fit at this point in the story.

Somehow I have to remember them all, keep them safe, ready to bring them back in when I *do* find their proper place in the story. Yet story is like that: the *telling* of the story may appear to be a nice, neat linear sequence - especially in hindsight - but it's often anything but linear when we're *within* the story itself. Tricky...

And then, moments later, that flood of ideas not only stops, but vanishes. Gone. A sense of flatness, nothingness, lostness; a sense of failure, almost. I don't know what to do... what do I do now? How can I recapture what just went past? How do I keep control of the story?

One of the hardest parts of the work of enterprise-architecture - especially when working with story - is that there *is* no control: 'control' is a myth, a fantasy, a delusion. Like story, it has its own time, which may not easily connect with ours. And it has its own imperatives, too, which likewise may conflict with ours. *Definitely* tricky there...

Application

- How do you capture ideas and images as they fly past in the moment? How do you find them again when you discover a place where they *do* fit within your enterprise-story? And how *do* you discover that ‘right place’ within the story?
- How do you cope with the ‘need’ for control of something that, almost by definition, is beyond any normal sense of control? How do you deal with *others’* need for you to seem to be in control of the story, or their need to be in ‘control’ of you in control of that story? What is it that *actually* holds the story together?

05: Circular stories

What kind of story-structure works best at the larger scale? If the Hero's Journey pattern is problematic here - because once we've achieved the goal, there's no more story - then what *do* we use?

For me, working with many different types of organisations over the past few decades, the answer seems to come in three distinct yet interleaved parts:

- the Story-Cycle
- the Strategy-Cycle
- the Market-Cycle

I'll describe each of those in turn, and how they interact, but perhaps not all of them in the one go.

Where this story starts is a long time ago - perhaps half a century or more? - with a man named Bruce Tuckman. He'd been trying to find a way to describe the sequence of activities in projects that succeed - and, equally, what happens in projects that don't. In particular, he recognised the importance of the people-issues, the 'group dynamics' in the lifecycle-story of the project.

And whilst the Hero's Journey does sort-of fit, the big difference is that projects are usually *collective*: it's more like an ensemble play than a follow-the-hero Hollywood

movie. What Tuckman noticed was that whilst everyone seems to want to rush in and get started straight away as soon as there's the first glimmering of a new idea, there are two crucial intermediate stages that have to happen first. And they have to happen in the right order, too. So here was his first version of that sequence:

- *Forming* - we develop the idea, intent and aim for the project
- *Storming* - we find the right people for the project, and deal with the interpersonal issues that invariably arise
- *Norming* - we settle down to do planning and preparation
- *Performing* - we do the work to deliver the project

If we skip the Forming stage, we end up with literally aimless action, 'doing for the sake of doing'. Not a good idea - especially in a business context.

If we skip the Storming stage, the project is likely to collapse in a squabbling heap right at the critical point - and it won't recover. Many people do find it difficult to deal with the sheer *messiness* of all the interpersonal stuff, but it *is* part of the work, whether we like it or not - and the project won't succeed without it.

And if we skip the Norming stage, we'll cripple the Performing stage, because the things that we need will turn

up in the wrong place, at the wrong time, be the wrong things, or be missing entirely. Whilst it's true that in many current project-contexts "no plan survives first contact with reality" (to paraphrase that old military dictum), we still need *some* form of plan - or *planning* - to ensure that we do have what we need when we need it.

Anyway, that's the core sequence for the project story: Forming, Storming, Norming, Performing.

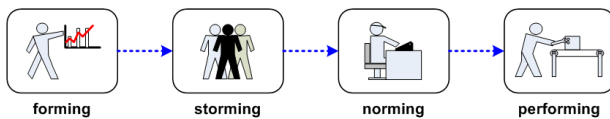


Figure 1: Tuckman sequence - linear

Yet whilst that story-structure makes sense for a single project, as viewed only in isolation, it's not quite complete in terms of what happens as one project ends and another begins. To deal with this, Tuckman added a final stage to the story-structure:

- *Adjourning* (or Mourning): we do an explicit completions and wrap-up for the project, including capture of lessons-learned

This then becomes a literal life-cycle, with the Adjourning of one project leading into and supporting the Forming of the next.

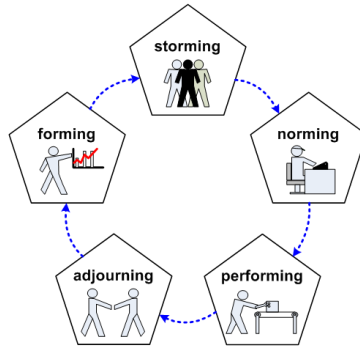


Figure 2: Tuckman sequence - cyclic

It's also a cycle that works in many different contexts - not just with projects or project-management - and at many different levels. To make the structure more generic, and to indicate that at the larger scale the stages may be handled by different parts of the organisation, I often use a different set of labels for this: Purpose, People, Preparation, Process, Performance. In essence, though, it's the same overall structure as in Tuckman's Group Dynamics.

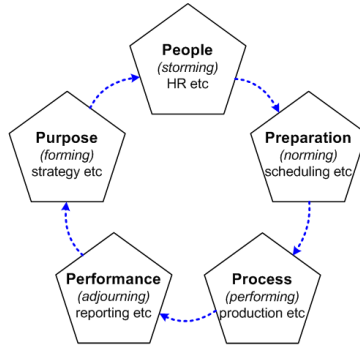


Figure 3: Five Elements cycle

Each area of emphasis also has a different *time*-perspective:

- *Purpose*: far-future
- *People*: ‘people-time’ or ‘story-time’ - anywhen from far-past to far-future, and from no-time to any-time
- *Preparation*: near-future
- *Process*: NOW!
- *Performance*: past

One of the concerns highlighted here is that it can be hard to link between Performance and Purpose, to close the cycle, because Purpose looks far into the future, whilst Performance looks only to the past. It’s the one place in

the cycle where there's such an extreme difference in time-perspective: and the problems can be hard to spot *because* it's at the apparent end of the sequence.

We do see similar mismatches of perspective, though, when we reframe this as the Strategy-Cycle, linking between strategy, tactics and execution. The Purpose and People phases focus more on 'feel'; Preparation and Performance alike focus on thinking, planning, analysis; whilst Process, of course, will focus most on the practice, the 'doing'. In most business-contexts, people tend to be very strong on the thinking and the doing, but often not at all comfortable with feeling - which can lead to serious plot-holes in the respective enterprise-story. But that's something we'll come back to later.

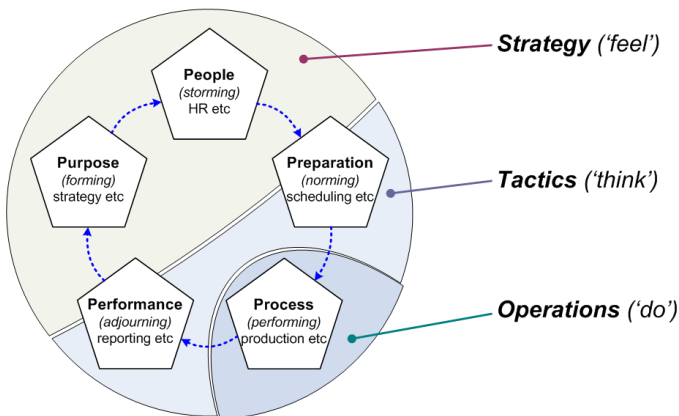


Figure 4: Strategy-cycle

This overall story-structure links well with cycles for continuous-

learning, such as Deming and Shewhart's PDCA (Plan, Do, Check, Act), whose Check phase aligns with Tuckman's Adjourning stage - though that's probably a separate story to here.

The Tuckman cycle also aligns almost perfectly with the classic Chinese *wu xing* or Five Element structure (Wood, Fire, Earth, Metal, Water). A lot of useful insights to be gained from the analogy - though that too is probably another story for elsewhen.

What *does* matter here, though, is the different kinds of leadership that drive the story forward. We not only need leadership to guide *within* each stage, we also need leadership to link from one stage to the next. And importantly, these are *different types of leadership*, requiring different skillsets, and hence often different people as the respective leader for each phase.

This means that our equivalent of the Hero's Journey here will need another set of phases, *between* each of the Tuckman-style stages in the story-lifecycle. For this I use an adaptation of Nigel Green's VPEC-T (Values, Policies, Event, Content, Trust): the original was about exchanges between entities, but reframing its core ideas in terms of lifecycles also works well. The difference is that a focus on Content is less relevant for *this* purpose, because it's implied everywhere, especially as part of the Process stage; instead, we need Completions, to mark the end-events of Process, and link to the next stage, to assess Performance.

Which, overall, gives us an enterprise-scope Story-Cycle,

that's also linked to the Strategy-Cycle:

- *Purpose* ['feel']: we develop the idea, intent and aim for the work - the *vision* for the work, and its expression as strategy
- *Values* ['feel']: we use vision and values embedded in the purpose to engage people's commitment to that purpose and strategy
- *People* ['feel']: we find the right people for the work, and deal with the interpersonal issues that invariably arise
- *Policies* [from 'feel' to 'think']: out of the discussion and arguments, we bring a framework for a plan and outline of tactics
- *Preparation* ['think']: we do all the planning and logistics needed to bridge between tactics and execution
- *Events* [from 'think' to 'do']: we await the trigger-event, the 'call to action'
- *Process* ['do']: we do the work to deliver the project
- *Completions* [from 'do' to 'think']: we note and act on the end-events that mark the call to *cease* action
- *Performance* ['think']: we do an explicit completions and wrap-up for the project, including capture of lessons-learned

- *Trust* [from ‘think’ to ‘feel’]: we connect the lessons-learned back to the initial aim and intent
- (and back to *Purpose* for the next iteration)

Or, in visual form:

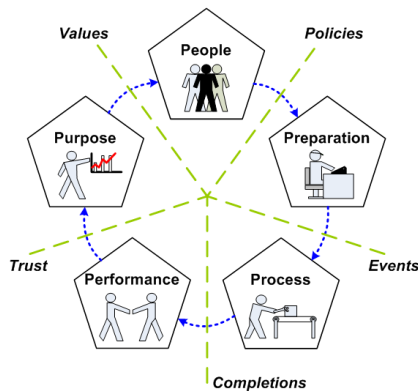


Figure 5: Story-cycle

There's one more type of leadership that's needed here, though it isn't part of the Story-Cycle itself: namely, the leadership needed to *hold the balance*. Another point we'll see again later, perhaps.

But right now, that's more than enough about the structure of story: this is supposed to be about story itself, after all. Time to heed that 'call to action', and move more into enterprise *as* story.

Application

- In your organisation's projects, where do you see examples of where Tuckman's sequence has *not* been followed: those five stages of Forming, Storming, Norming, Performing and Adjourning either occurring out of sequence, or skipped entirely? What are the consequences when this occurs?
- What methods are used within your organisation to capture and apply lessons-learned from projects? Within your organisation and your architecture, how do you link one project to the next? What are the stories here?
- At the larger scale, which parts of your organisation specialise in Purpose? (Strategy, for example?) Which parts emphasise the People-issues? Preparation? Process, or production? The tracking of Performance, and lessons-learned? Who leads each of these areas of the organisation?
- Which parts of your organisation link *between* each of those specialist emphases? Who leads that linking between areas, and how?
- What issues arise for your organisation because different areas have different time-perspectives? In what ways can you use architecture, and story-as-architecture, to bridge across those different world-views?

- What happens to the balance of the whole when one area dominates over all the others? What can you do in *architectural* terms to bring the overall story back into balance?
- What do you see when you apply the Strategy-Cycle and Story-Cycle to example areas of your organisation? What's missing? Which stages are being skipped, or scrambled? In what stages of the cycle are there mixed-up mixes of 'feel', 'think' and 'do'? What are the consequences when this happens? And what can you do in *architectural* terms, with structure or with story, to bring it into better balance?

20: Wrap up the story

Time to get back to the everyday world, I guess: there's a backlog of work that I need to get back to, and it won't wait around any longer. Sigh...

Some story, though. A story about story, that itself is a story.

The storytellers often talk at this point about 'the return with the elixir': returning from the sojourn in the 'special-world' with something new, something different, something that that makes a real difference for everyone. Somehow story seems to be a lot *quieter* than that: I don't know that I'd make any of it out to be so special, yet for me it's true there'd be a fair few themes here that might almost fit as 'the elixir of story':

- architecture is about structure, yet it's *also* about story
- story is always about *people*, in one sense or another
- story is what engages people in the aims of the enterprise - whatever that enterprise may be
- narrative provides a sequence of action in context - it becomes a story only when we add the *why*
- organisation focuses on structure, yet the enterprise *is* the story - the structure happens *because* of the story

- structures may be re-used in other stories - but the structure *itself* is not the story
- those random-seeming interludes and asides *are* part of the story - and often provide the most useful side of that story
- we don't control the story, and we don't possess it - it's more like *it* possesses *us*
- the story's over when it's over - and then it often starts again anyway

Just like structure, story is *everywhere* in enterprise-architecture - and just as important, too. Yet also just like structure, story isn't everything: it's just a story. We can use it, or not, as we need.

And that's really the point, perhaps: story is nothing special - it's just another tool in the enterprise-architect's toolkit. Might not have noticed it much before, but it's there now when we need it, and - like any good tool - it's useful when used in the right way.

So what would we do different on Monday morning, as a result of all of this? Maybe quite a lot, maybe nothing at all: it's up to each of us, really. Working with story can be a very different way of working: at the very least, it'll take some time to seep in to our way of *seeing*, the way we think about architectures in general. If we want, we can choose to stick with a classic structure-only view: there's nothing wrong in that, it's just a bit more limiting, that's

all. If we do that, though, at least we do now know that the story-side is there and ready for us whenever we need it - and to make our architecture-work a lot more interesting, too.

See what happens on Monday morning, I guess: a different kind of story, perhaps?

Application

- What for you would be ‘the elixir of story’ for enterprise-architecture? What themes stood out for you as you read through this book?
- Which items could you use straight away in your enterprise-architecture toolkit? Which items would need more practice, or wait until some appropriate context comes along?
- And which items still seem to make no sense at all? In what ways are you willing to trust that they *will* make sense when the time is right, even if not before? What’s the story behind that story?
- Given what you’ve seen here, what would you do differently come Monday morning, in your enterprise-architecture or elsewhere? And what stories do you need - or what *different* stories do you need - to ensure that you *can* do it differently on Monday morning?

Appendix: Sources and resources

Sources

This is a summary of the main references and sources mentioned in the text above. It's in alphabetical order, with books sorted by title rather than author.

- Matthew Frederick, *101 Things I Learned in Architecture School* (MIT Press, 2007)
- Archimate modelling-notation for enterprise-architecture: see Wikipedia, en.wikipedia.org/wiki/ArchiMate¹ and Open Group, www.opengroup.org/archimate²
- Business Model Canvas: see Wikipedia, en.wikipedia.org/wiki/Business_Model_Canvas³; also Alex Osterwalder, Yves Pigneur et al., *Business Model Generation: a handbook for visionaries, game-changers and challengers* (self-published, 2010)
- Christopher Alexander, *A Pattern Language: Towns, Buildings, Construction* (Oxford University Press Press, 1977); also see Wikipedia, en.wikipedia.org/wiki/A_Pattern_Language⁴

¹<http://en.wikipedia.org/wiki/ArchiMate>

²<http://www.opengroup.org/archimate>

³http://en.wikipedia.org/wiki/Business_Model_Canvas

⁴http://en.wikipedia.org/wiki/A_Pattern_Language

- BPMN (Business Process Modeling Notation): see Wikipedia, en.wikipedia.org/wiki/BPMN⁵
- Chris Crawford, *Chris Crawford on Interactive Storytelling* (New Riders, 2005)
- Cluetrain Manifesto: see www.cluetrain.com⁶
- ‘Deadly-embrace’ in systems-design: see Wikipedia, en.wikipedia.org/wiki/Deadlock⁷
- DoDAF ([US] Department of Defense Architecture Framework): see Wikipedia, en.wikipedia.org/wiki/DoDAF⁸
- Dramatica story-structure: see dramatica.com⁹ and storymind.com/dramatica¹⁰; also Armando Saldaña-Mora, *Dramatica for Screenwriters* (Write Brothers, 2005)
- Enterprise as story (on Tetradian weblog): weblog.tetradian.com/the-enterprise-is-the-story¹¹
- Enterprise Canvas: see Tom Graves, *Mapping The Enterprise: modelling the enterprise as services with the Enterprise Canvas* (Tetradian Books, 2010); also summary at tetradianbooks.com/ecanvas-summary¹²

⁵<http://en.wikipedia.org/wiki/BPMN>

⁶<http://www.cluetrain.com>

⁷<http://en.wikipedia.org/wiki/Deadlock>

⁸<http://en.wikipedia.org/wiki/DoDAF>

⁹<http://dramatica.com>

¹⁰<http://storymind.com/dramatica>

¹¹<http://weblog.tetradian.com/the-enterprise-is-the-story>

¹²<http://tetradianbooks.com/ecanvas-summary>

- eTOM / Frameworkx: see Wikipedia, en.wikipedia.org/wiki/Enhanced_Telecom_Operations_Map¹³ and en.wikipedia.org/wiki/Frameworkx¹⁴
- FEAF ([US] Federal Enterprise Architecture Framework): see Wikipedia, en.wikipedia.org/wiki/Federal_Enterprise_Architecture¹⁵
- Five Elements (*wu xing*): see Wikipedia, en.wikipedia.org/wiki/Wu_xing¹⁶
- Hero's Journey story-structure: see www.thewritersjourney.com/hero_journey.htm¹⁷
- Film *Kate & Leopold*: initial version of screenplay, www.weeklyscript.com/Kate%20And%20Leopold.txt¹⁸; summary of final-version plot on Wikipedia, [en.wikipedia.org/wiki/Kate_Leopold](http://en.wikipedia.org/wiki/Kate_Leopold)
- Len Fehskens at The Open Group: see www.opengroup.org/contacts/bio.htm¹⁹
- 'Manifesto' on power and responsibility in the workplace [PDF]: tetradianbooks.com/ebook/hss-manifesto.pdf²⁰

¹³http://en.wikipedia.org/wiki/Enhanced_Telecom_Operations_Map

¹⁴<http://en.wikipedia.org/wiki/Frameworkx>

¹⁵http://en.wikipedia.org/wiki/Federal_Enterprise_Architecture

¹⁶http://en.wikipedia.org/wiki/Wu_xing

¹⁷http://www.thewritersjourney.com/hero's_journey.htm

¹⁸<http://www.weeklyscript.com/Kate%20And%20Leopold.txt>

¹⁹http://www.opengroup.org/contacts/bios/fehskens_bio.htm

²⁰<http://tetradianbooks.com/ebook/hss-manifesto.pdf>

- Andrew McAfee and ‘It’s not not about the technology’: see andrewmcafee.org/2007/its_not_not_about_the_technology²¹
- Memory-theatre and ‘the art of memory’: see Wikipedia, en.wikipedia.org/wiki/Art_of_memory²²; also Frances Yates, *The Art of Memory* (Routledge Kegan Paul, 1966)
- Narrative-paradigm (Walter Fisher): see Wikipedia, en.wikipedia.org/wiki/narrative_paradigm²³
- PDCA (plan, do, check, act): see Wikipedia, en.wikipedia.org/wiki/PDCA
- John Hagel III, John Seely Brown and Lang Davison, *The Power of Pull: How Small Moves, Smartly Made, Can Set Big Things in Motion* (Basic Books, 2010)
- Chris Potts, *recreAtion: realizing the extraordinary contribution of your enterprise-architects* (Technics, 2010)
- SCOR (Supply-Chain Operations Reference) framework: see Wikipedia, en.wikipedia.org/wiki/SCOR²⁵
- Gerald M Weinberg, *The Secrets of Consulting: a guide to giving and getting advice successfully* (Dorset House, 1986) and *More Secrets of Consulting: the*

²¹http://andrewmcafee.org/2007/its_not_not_about_the_technology

²²http://en.wikipedia.org/wiki/Art_of_memory

²³http://en.wikipedia.org/wiki/narrative_paradigm

²⁴<http://en.wikipedia.org/wiki/PDCA>

²⁵<http://en.wikipedia.org/wiki/SCOR>

consultant's toolkit (Dorset House, 2002); see also www.geraldmweinberg.com²⁶

- Story-listening: see Anecdote white-paper, *Making the most of story*, www.anecdote.com.au/whitepapers.php?wpid=27²⁷
- Strategy-story and business use of story: see Anecdote, www.anecdote.com²⁸
- Lao Tsu (tr. Gia Fu Feng and Jane English), *Tao Te Ching* (Wildwood House, 1973)
- The 'This'-game (requirements-elicitation): see weblog.tetradian.com/exploratory-game-for-service-oriented-ea²⁹ and weblog.tetradian.com/on-the-this-game-for-ea³⁰
- TOGAF (The Open Group Architecture Framework): www.opengroup.org/togaf³¹
- TRAK (architecture framework for London Underground and [UK] Department of Transport): see Wikipedia, en.wikipedia.org/wiki/TRAK³²
- Transmedia and transmedia-storytelling: see Wikipedia, en.wikipedia.org/wiki/Transmedia_storytelling³³

²⁶<http://www.geraldmweinberg.com>

²⁷<http://www.anecdote.com.au/whitepapers.php?wpid=27>

²⁸<http://www.anecdote.com>

²⁹<http://weblog.tetradian.com/this-exploratory-game-for-service-oriented-ea>

³⁰<http://weblog.tetradian.com/more-on-the-this-game-for-ea>

³¹<http://www.opengroup.org/togaf>

³²<http://en.wikipedia.org/wiki/TRAK>

³³http://en.wikipedia.org/wiki/Transmedia_storytelling

- Tuckman Group Dynamics: see Wikipedia, en.wikipedia.org/wiki/Forming-storming-norming-performing³⁴
- UML (Unified Modeling Language): see Wikipedia, en.wikipedia.org/wiki/Unified_Modeling_Language³⁵
- United Breaks Guitars: see Wikipedia, en.wikipedia.org/wiki/United_Breaks_Guitars³⁶
- VPEC-T: see Wikipedia, en.wikipedia.org/wiki/VPEC-T³⁷; also Nigel Green and Carl Bate, *Lost in Transition: a handbook for information-systems in the 21st century* (Evolved Technologist Press, 2007)
- Walmart and sustainability: see [](<http://walmartstores.com/sustainability>)
- Wicked-problem: see en.wikipedia.org/wiki/Wicked_problem³⁸
- Zachman Framework: see Wikipedia, en.wikipedia.org/wiki/Zachman_Framework³⁹

³⁴<http://en.wikipedia.org/wiki/Forming-storming-norming-performing>

³⁵http://en.wikipedia.org/wiki/Unified_Modeling_Language

³⁶http://en.wikipedia.org/wiki/United_Breaks_Guitars

³⁷<http://en.wikipedia.org/wiki/VPEC-T>

³⁸http://en.wikipedia.org/wiki/Wicked_problem

³⁹http://en.wikipedia.org/wiki/Zachman_Framework

The Tetradian Enterprise Architecture series

The Tetradian Enterprise Architecture series of books present new developments on theory, principles and practice of enterprise-architecture - moving beyond IT to the whole enterprise.

- Tom Graves, *Real Enterprise-Architecture: beyond IT to the whole enterprise* (Tetradian Books, 2008)
- Tom Graves, *Bridging the Silos: enterprise-architecture for IT-architects* (Tetradian Books, 2008)
- Tom Graves, *SEMPER SCORE: enhancing enterprise effectiveness* (Tetradian Books, 2008)
- Tom Graves, *Power and Response-ability: the human side of systems* (Tetradian Books, 2008)
- Tom Graves, *The Service-Oriented Enterprise: enterprise architecture and viable services* (Tetradian Books, 2009)
- Tom Graves, *Doing Enterprise Architecture: process and practice in the real enterprise* (Tetradian Books, 2009)
- Tom Graves, *Everyday Enterprise Architecture: sense-making, strategy, structures and solutions* (Tetradian Books, 2010)

- Tom Graves, *Mapping the Enterprise: modelling the enterprise as services with the Enterprise Canvas* (Tetradian Books, 2010)

Other resources

The following are some other links and references that you may find useful in terms of enterprise-as-story:

- Barely Repeatable Processes: see www.thingamy.com⁴⁰
- BMM (Business Motivation Model): see businessrulesgroup.org/bmm.shtml⁴¹
- Causal Layered Analysis: see Wikipedia, en.wikipedia.org/wiki/Causal_layered_analysis⁴²
- Living organisation: see Arie de Geus, *The Living Company: Habits for Survival in a Turbulent Business Environment* (HBR Press, 2002)
- OODA (observe, orient, decide, act): see Wikipedia, en.wikipedia.org/wiki/OODA_loop⁴³
- Porter Value-Chain: see Wikipedia, en.wikipedia.org/wiki/Value_chain⁴⁴

⁴⁰<http://www.thingamy.com>

⁴¹<http://businessrulesgroup.org/bmm.shtml>

⁴²http://en.wikipedia.org/wiki/Causal_layered_analysis

⁴³http://en.wikipedia.org/wiki/OODA_loop

⁴⁴http://en.wikipedia.org/wiki/Value_chain

- RACI (responsible, assists, consulted, informed): see Wikipedia, en.wikipedia.org/wiki/Responsibility_assignment_matrix⁴⁵
- Shell General Business Principles: www.shell.com/sgbp⁴⁶
- Value-stream mapping: see Wikipedia, en.wikipedia.org/wiki/Value_stream_mapping⁴⁷
- VRMG (vision, role, mission, goal): see www.slideshare.net/tetradian/role-mission-goal-a-framework-for-business-motivation⁴⁸

⁴⁵http://en.wikipedia.org/wiki/Responsibility_assignment_matrix

⁴⁶<http://www.shell.com/sgbp>

⁴⁷http://en.wikipedia.org/wiki/Value_stream_mapping

⁴⁸<http://www.slideshare.net/tetradian/vision-role-mission-goal-a-framework-for-business-motivation>