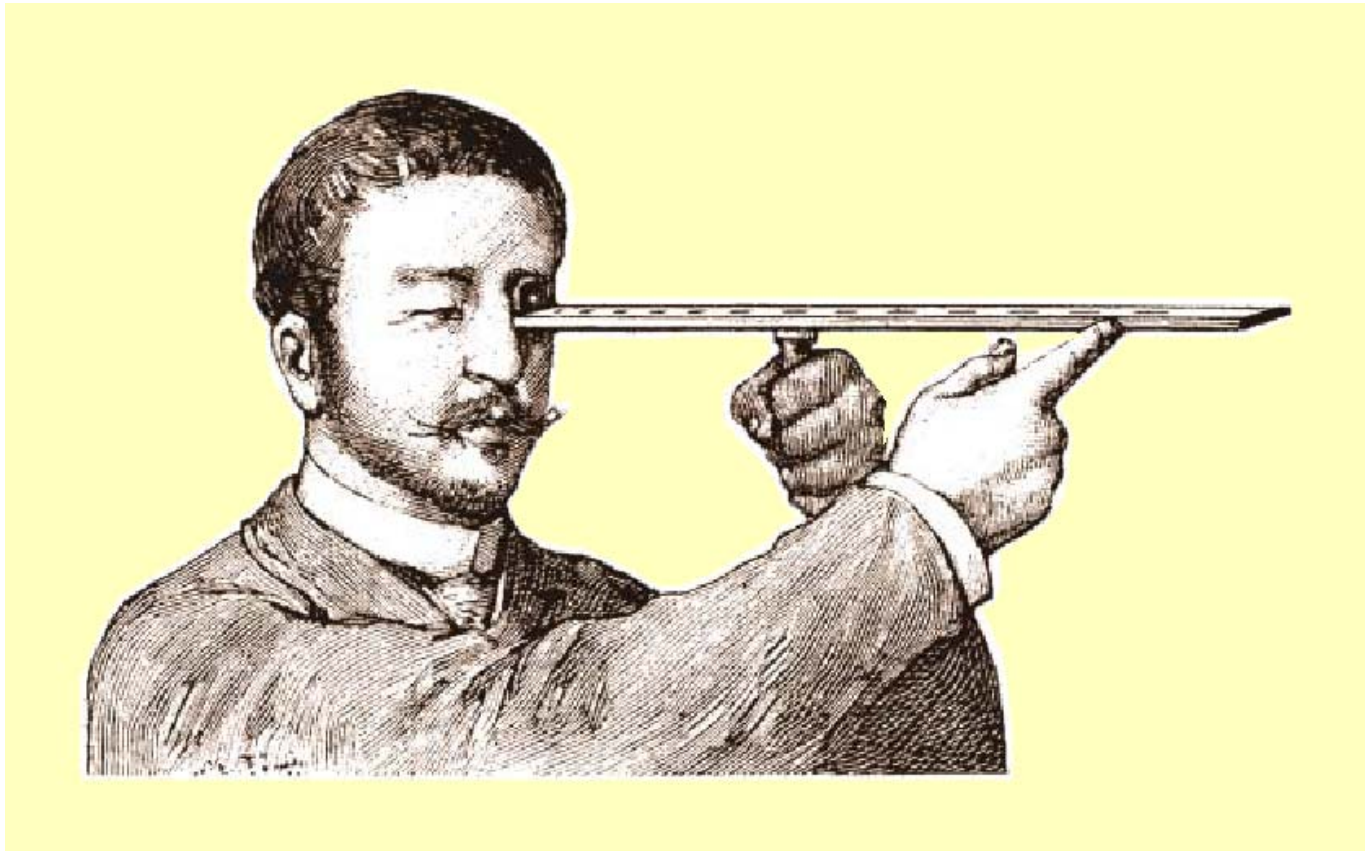


Contents and sample chapters from

Real Enterprise-Architecture

Beyond IT to the whole enterprise



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>

Initial release (e-book only) by Tetradian Consulting: April 2007

First published: May 2008

ISBN 978-1-906681-00-5 (paperback)

ISBN 978-1-906681-01-2 (e-book)

© Tom Graves 2008

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.

Contents [Sample chapters]

- An overview 1
- DA :: The aims of architecture 11
- DN :: Architecture of the enterprise 14
- DR :: The architecture of the everyday 15
- DE :: Architecture on purpose 16
- DL :: Architecture is a feeling 18
- PL :: The architecture team 19
- PA :: The politics of purpose..... 28
- PR :: A problem of power..... 29
- PN :: The role of the generalist 30
- PE :: What’s the story? 31
- KE :: Dimensions of architecture..... 32
- KN :: An emphasis on effectiveness 33
- KA :: Architecture as a way of thinking..... 37
- KL :: A question of responsibility 38
- KR :: The centrality of services 39
- TE :: Requirements for agility 40
- TN :: Managing services 41
- TR :: The practice of architecture 42
- TL :: The art of integration 43

TA :: What's the SCORE?	48
MR :: Real-time scoreboards	49
ME :: Closing the loop.....	50
ML :: People and performance	51
MA :: Measuring maturity	52
MN :: Monitoring integration.....	53
Glossary	55

Acknowledgements

Amongst others, the following people kindly provided comments and feedback on the early drafts of this book: Daljit Banger (White Knight Consulting, UK), Bill Branson (Whiteboards That Work, US), Shawn Callahan (Anecdote, Aus), Adrian Campbell (Ingenia, UK), Jane Chittenden (Format Design, UK), Roland Ettema (LogicaCMG, NL), Sandra Fields (NSW DoCS, Aus), John Forrest (Holocentric, Aus), Ken Harper (Gartner, UK), James McGovern (ArchitectBook, US), Helen Mills (Australia Post, Aus), Robert Phipps (NSW DoCS, Aus), Liz Poraj-Wilczynska (Brockhampton, UK), Roy Roebuck (One World Information System, US), Gerry Ryan (NSW DoCS, Aus).

Please note that, to preserve commercial and personal confidentiality, the stories and examples in this book 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 organisation or individual.

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

AN OVERVIEW

About this book

As with many writers, this project arose out of frustration that the book I needed for my work didn't exist. There were plenty about the minutiae of the field I work in – called 'enterprise architecture' – and about some of the frameworks that can be used for small subsets of the work – mainly in IT – but nothing that really covered the full scope.

Over the years, that frustration grew and grew. Until finally I succumbed to what Open Source programmers describe as 'the need to scratch an itch'. And sat down to write.

So this book is about the practice of enterprise-architecture, particularly at the level of the whole enterprise. And it's for anyone who works with the enterprise as a whole: chief officers, strategists, programme management office – roles of that kind.

But what *is* enterprise-architecture, anyway?

What is this thing called enterprise-architecture?

Many business-folk have never heard of enterprise-architecture. Which is not surprising, because most of the literature in the field suggests it's about IT, and *only* about IT. There might be a few throwaway references somewhere to some blurry notion of 'business architecture', but that's about it. Hence of no relevance to everyday business, really.

Which is a problem, because *real* enterprise-architecture isn't much about IT at all. Or rather, although IT is significant, it's only one small part. Turns out instead that that blurry 'business architecture' isn't something that can be skipped over in a rush down to the technical minutiae: it's the *core* of enterprise architecture.

Enterprise-architecture is about the architecture – the structure – of the *whole* of the enterprise:

Enterprise-architecture is the integration of everything the enterprise is and does.

Even the term ‘architecture’ is perhaps a little misleading. It’s on a much larger scale, the scale of the whole rather than of single sub-systems: more akin to city-planning than to the architecture of a single building. In something this large, there are no simple states of ‘as-is’ versus ‘to-be’, because its world is *dynamic*, not static. And it has to find some way to manage the messy confusion of what *is*, rather than the ideal that we might like it to be.

The gym I frequent in this ancient garrison town is housed in a former NAAFI – a military commissary and entertainment centre, in US terms. It’s a relic of the Second World War – and looks it, too. Scattered around in the battered old building there’s a bar, a ballroom, a large café, a hairdressing school, a tanning salon, and an ever-changing variety of small clubs and business oddities. Stairs and passageways wander off at random, with side-rooms that don’t seem to be used at all; wouldn’t be surprised if the keys had been lost for decades.

In short, a mess. But *somehow* it does all work as a whole. Sort of.

Much like most large businesses, in fact. Which is why we need enterprise-architecture.

Grandiose plans won’t help us much in making sense of all this complexity. What we need is something simple that will allow us to start small, yet keep a consistent structure as we expand the scope upward to the whole of the enterprise and its environment. So what we need is a *system* for enterprise architecture.

A systematic approach

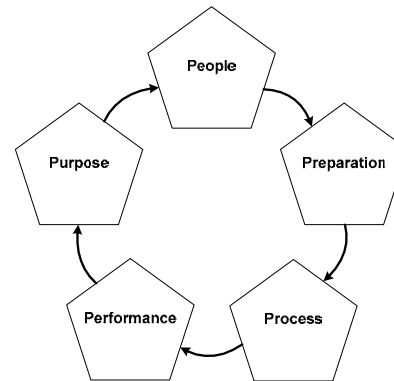
As a discipline, IT-architecture has been around for the past twenty years or so. So there are a fair few frameworks available, all of them good: Zachman, FEAF, TOGAF and ARIS, to name some of the better-known examples. (You’ll find links to these in the Resources section at the end of the chapter.).

But what they’re good at, unfortunately, is *IT-architecture* – not *enterprise-architecture*, in this broader sense. Paradoxically, to go wider, we need something simpler.

The framework I’ll use here is about as simple as it gets. We start with a ‘5Ps’ variant of the old Group Dynamics project life-cycle, with *Purpose*, then *People*, *Preparation*, *Process* and *Performance*:

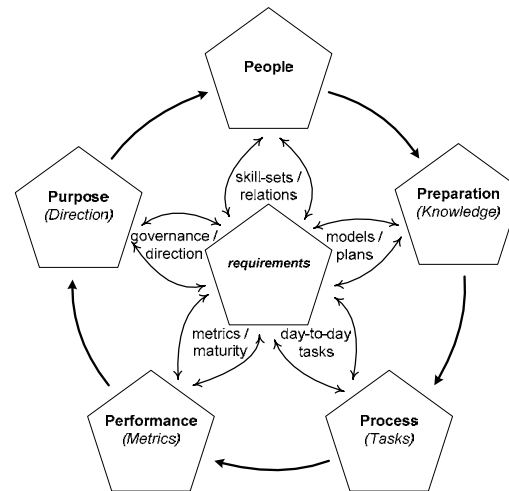
- **Purpose** is about the *why* of business – beginnings, intentions, strategy, direction.

- **People** is about the *who*, the teams and skill-sets and relationships we need to put the strategy into action.
- **Preparation** is about *what*, and *where*, and *when*, and *how*: it grounds the ideas and arguments into a concrete plan.
- **Process**, or Practice, puts that plan into action.
- **Performance** assesses the results – not just of the Process, but of the integration of the whole – feeding back into the Purpose for a new cycle.



5Ps project-lifecycle framework

Each of these points in the cycle is also a perspective onto the whole, and provides shared requirements that impact on the rest of the enterprise. Mapping these in turn to the issues and artefacts that would be the concern of enterprise-architecture, we find ourselves with a framework that looks like this:



5Ps framework with enterprise-architecture artefacts

If you're familiar with the Open Group Architectural Framework (TOGAF), you might recognise some similarities with the cycle shown in TOGAF's Architectural Design Method (ADM). What's different here is that there's no special emphasis given to IT: it's just one part of a much larger whole, almost all of which the ADM bundles into an arbitrary, undifferentiated box labelled 'Business Architecture'. In essence, what this framework does is unpack that box and put everything into its proper perspective.

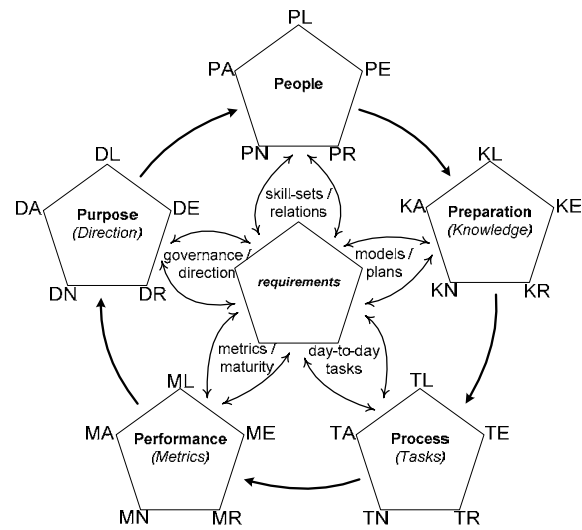
We apply a systems-principle called *recursion*, and get each view to look at each of the others, and at the integration of the whole. In other words, each view also contains within itself a sense of all the others. This gives us a means to assess *effectiveness* – the overall impact of every part of the enterprise on everything else.

Although they're similar to those '5Ps', we give these sideways views a slight twist, to give us the keywords *Efficient, Reliable, Elegant, Appropriate* and *Integrated*. This gives us a set of views-within-views, which we'll label with two-letter codes as follows:

	<i><u>E</u>fficient</i>	<i><u>R</u>eliable</i>	<i><u>E</u>legant</i>	<i><u>A</u>ppropriate</i>	<i><u>I</u>ntegrated</i>
<i>Purpose (<u>D</u>irection)</i>	DE	DR	DL	DA	DN
<i><u>P</u>eople</i>	PE	PR	PL	PA	PN
<i>Preparation (<u>K</u>nowledge)</i>	KE	KR	KL	KA	KN
<i>Process (<u>T</u>asks)</i>	TE	TR	TL	TA	TN
<i>Performance (<u>M</u>etrics)</i>	ME	MR	ML	MA	MN

which gives us an overall framework which may look abstract at first, but takes very little effort to translate it into practice. And its consistency and symmetry make it easy to apply across the full scope of the enterprise – which can't be done easily, or at all, with many of the existing frameworks. We'll also see later why and how this kind of iterative, recursive structure supports enterprise agility, usually at a much lower cost than conventional IT-centric forms of enterprise-architecture.

But for now, this provides us with the skeleton for the book: one section for each of the five focus-types, each with five short chapters emphasising specific aspects of effectiveness in relation to enterprise-architecture.



5Ps architecture framework

And each chapter, in turn, has a section on the principles underlying the issue, followed by suggestions on how to apply those principles in practice, using the same sequence as the main pattern: purpose, people, preparation, process, performance.

Each chapter ends with a brief section on broader usage of the same perspective beyond enterprise-architecture, and includes a list of some suggested additional resources.

‘D’ group: Purpose - Direction

This is about the far-future focus of business-direction and strategy – typically the preserve of management.

For enterprise-architecture, we need to address the purpose of architecture itself in relation to the enterprise:

- *DA :: The aims of architecture* explores the business drivers – the need to reduce cost and complexity, and to support enterprise integration and agility, that underpin the need for an enterprise-architecture.

- *DN :: Architecture of the enterprise* reviews the development of enterprise-architecture from its roots in IT, and the issues that drive the need to break away from those roots.
- *DR :: The architecture of the everyday* illustrates the need for enterprise-architecture to be relevant to everyday practice, assisting in the day-to-day decisions at the 'coal-face'.
- *DE :: Architecture on purpose* explores the 'audit-trail' of vision, role, missions and goals which provide the anchor for business motivation – and for the enterprise-architecture.
- *DL :: Architecture is a feeling* reminds us that an enterprise is made up of people – so we need to support values, and community, as an explicit part of the enterprise-architecture.

'P' group: People

In the enterprise itself, this is the complicated, almost outside-of-time concerns that people bring to an enterprise.

These are the everyday affairs of HR and the 'people people', but we also need to take them into account in enterprise architecture:

- *PL :: The architecture team* addresses the nature of the enterprise-architecture team itself: skill-sets, mindsets and overall attitudes to the work and to each other.
- *PA :: The politics of purpose* reminds us that everything in an enterprise is underpinned by the complex politics of change.
- *PR :: A problem of power* tackles the subtle balance of power and responsibility upon which the successful implementation of the enterprise-architecture will depend.
- *PN :: The role of the generalist* identifies the difficulties faced by those whose often unnoticed work provides the bridges between organisational 'silos'.
- *PE :: What's the story?* focuses on the narrative knowledge shared by and between people, through which most business meaning is derived.

'K' group: Preparation - Knowledge

This is the near-future emphasis of bringing things and information together to support an upcoming activity – the realm of planners, schedulers, logistics and similar roles.

For enterprise-architecture, it's also about the ideas and images which provide the theoretical foundations for its practice:

- *KE :: Dimensions of architecture* takes a more in-depth look at the business dimensions that underpin the framework.
- *KN :: An emphasis on effectiveness* expands on those 'views' for assessing effectiveness: efficient, reliable, elegant, appropriate, integrated.
- *KA :: Architecture as a way of thinking* introduces the strange realms of system-theory and their immediate practical applications for the agile enterprise.
- *KL :: A question of responsibility* addresses the importance of responsibility-based 'ownership' for the non-tangible assets such as projects, business data and business rules.
- *KR :: The centrality of services* expands the core concept of 'service' in a service-oriented architecture.

'T' group: Process – Tasks

At the point of production, there's a necessary focus on the urgency of the "now!".

The same applies to enterprise-architecture: its practices must support the immediacy of everyday business concerns:

- *TE :: Requirements for agility* surveys some of the Agile methodologies such as XP and DSDM, and how to support these through requirements linked to the enterprise-architecture.
- *TN :: Managing services* describes the role of complementary frameworks such as ITIL, TQM and COBIT, and how to integrate these into a broader enterprise-architecture.
- *TR :: The practice of architecture* goes into more detail on the iterative, recursive process of creating and managing compliance to an enterprise-architecture.
- *TL :: The art of integration* looks at how to use a unifying theme, such as privacy, quality, trust or waste-reduction, in parallel with enterprise-architecture, to increase integration across the enterprise.
- *TA :: What's the SCORE?* introduces a strategic tool to assess potential impacts on overall effectiveness across the enterprise.

‘M’ group: Performance – Metrics, artefacts and outcomes

In business, this provides the ‘rearward view’ looking back at completions, deliverables, outcomes of activities – concerns such as sales-fulfilment, after-market monitoring, accounts receivable, statistics and the like.

This is about how the enterprise architecture assists in monitoring enterprise performance – and also the performance of the architecture itself:

- *MR :: Real-time scoreboards* investigates sources for key information in a ‘balanced scorecard’ to track enterprise performance.
- *ME :: Closing the loop* explores the mechanisms needed for feedback from other areas into the enterprise-architecture, using tools such as After Action Reviews.
- *ML :: People and performance* describes a method to identify ‘the ability to do work’ across the enterprise, and how to use that information to develop interventions as appropriate.
- *MA :: Measuring maturity* is about the capability of the enterprise-architecture itself, and what needs to be done at each stage to expand its potential.
- *MN :: Monitoring integration* summarises the frameworks and metrics needed to monitor impact of enterprise-architecture on whole-of-organisation integration.

Glossary

The last part of the book provides some additional resources such as a brief glossary of terms.

Using the framework

The aim of this book is that you can apply the framework immediately to your own area of work, and begin to see useful results within a matter of days. Filling in the *detail* of the skeleton will take weeks, months, years, of course: but the skeleton alone is enough to get started on a new view of the enterprise as a whole. Once the skeleton is in place, the detail can be developed by working iteratively, recursively, through the framework in whatever area and to whatever level may be needed at the time. In other words, a “just enough, just in time” approach to enterprise architecture.

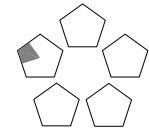
Applying the same principle of recursion, this book itself illustrates the framework in action. I've used the same approach as the Agile methodology DSDM – features are limited by time-budget – which in this case has meant allowing myself just two months to get down on paper as much as I can from a couple of decades of work. So I make no pretence that this is some “complete, ultimate guide” to enterprise architecture: it's just one iteration for a much larger, shared body of knowledge – to which someday I hope you will contribute.

But first, read once, quickly, through the whole book; then dip into the framework's toolbox at random, as needed. I'd recommend that you clarify your purpose (Chapters DA to DL) and set up some kind of requirements methodology (chapter TE) before you start; but from then on, almost any order will do. Enjoy!

Resources

- 🔖 Zachman framework: <http://www.zifa.com>
- 🔖 DyA (Dynamic Architecture): <http://eng.dya.info/Home/>
- 🔖 TOGAF framework: see <http://www.opengroup.org/architecture/togaf8-doc/arch/toc.html>
- 🔖 FEAF framework: <http://www.cio.gov>
- 🔖 ARIS methodology: <http://www.ids-scheer.com/international/English/products/53961>
- 🔖 ArchiMate enterprise-architecture notation: <http://www.telin.nl/index?cfm.project=ArchiMate&language=en>

DA ::THE AIMS OF ARCHITECTURE



Principles

What are the aims of the architecture?

This one's straightforward, and usually short and sweet, but it does need to be done *first* - otherwise there's a high risk that time and effort would be wasted. You'll also need to come back and review the results from time to time, for the same reason.

What we establish here are people's expectations of the enterprise architecture. If the aim is to cut costs, for example, or support agility, or reduce complexity, each expectation needs to be documented. This then gives you a list of 'key success criteria' from which to work. *How* you'll achieve such success is something we'll address later, in practice: for now, all we need is that list.

Procedure

Purpose

Identify the overall purpose of the organisation's enterprise architecture.

People

Senior managers, strategists, enterprise-architects.

Preparation

Standard business-analysis tools: whiteboard, meeting-space, pen and paper, time, and access to the required people.

Process

Use standard business-analysis interviews and group facilitation with key stakeholders such as senior managers, to elicit a list of key success criteria.

These are likely to arise first as simple expectations or desires:

- rein in the cost and complexity of IT systems
- support collaboration across the enterprise
- change to a more customer-centric enterprise
- increase adaptability and agility in the marketplace
- improve management of risk and opportunity
- simplify and speed up integration of mergers and acquisitions

For a first iteration, the list above is probably enough: it gives *something* to start with.

In later iterations, you'll need more explicit answers. For each of the 'desires' in the list, ask clarifying questions such as "How will we know when we've achieved this?" and "What would you regard as inadequate / adequate / good / excellent success in relation to this?". Often one desire will be dependent on another: for example, agility is unlikely to happen without reduced complexity. And at the *enterprise* level, success may well depend on interactions across the whole enterprise, making standard silo-based performance measures almost meaningless. What we're after here is 'key performance indicators' and success-criteria that arise from the *whole*, not necessarily from any one part.

These success-criteria become a core part of the enterprise-architecture charter: the measure of its mission, its "capability that will be achieved and maintained indefinitely thereafter".

Performance (artefacts and outcomes)

List of core requirements and key success criteria (KSCs) for enterprise architecture and whole-of-organisation integration.

Broader applications

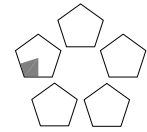
This perspective is *Purpose / Appropriate* – a recursive emphasis on the purpose of business-purpose itself. To structure the organisation's knowledge about this, see *DE :: Architecture on purpose* (p.16); to explore the emotional base, see *DL :: Architecture is a feeling* (p.18); but here this is more about what the organisation *is* – its identity and reason-to-be.

Does the identity match the purpose? Conversely, does the purpose match the identity? Conventional marketing-style analysis and corporate-identity development would be useful here, though if the organisation's power-dynamics will permit it – see *PR :: A problem of power* (p.29) – more value may be gained through large-group techniques such as Open Space or Future Search, or participative depth analysis methodologies such as Causal Layered Analysis, to access and identify the organisation's foundational 'myths'.

Resources

- 🏠 Large group interventions: see Martin Leith, www.largegroupinterventions.com/documents/leiths_guide_to_lgis.pdf
- 🏠 Depth analysis: see Sohail Inayatullah, www.metafuture.org/Articles/CausalLayeredAnalysis.htm

DN :: ARCHITECTURE OF THE ENTERPRISE



Principles

Having defined the aims of the architecture, we next need to be clear about its scope: just how much of the enterprise will our 'enterprise architecture' address?

This is not as simple as it sounds, because if we're not careful, this innocent-seeming question can lead us into some very murky waters on the shores of organisational politics.

[[see published book for details]]

Procedure

[[see published book for details]]

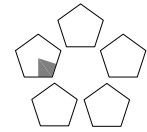
Broader applications

[[see published book for details]]

Resources

[[see published book for details]]

DR :: THE ARCHITECTURE OF THE EVERYDAY



Principles

What's the purpose of what you will actually *do*? How will the architecture activities help to drive the day-to-day decisions of the enterprise?

This section isn't about the detail of architecture activities – that'll be addressed in *TR :: The practice of architecture* (p.42) – but about the aims and expectations of the architecture's 'customers'. In a sense we'll be covering some of the same ground as in *DA :: The aims of architecture* (p.11), but from the opposite direction: 'bottom-up', from the perspective of individual work-teams, rather than 'top-down', for the enterprise as a whole.

[[see published book for details]]

Procedure

[[see published book for details]]

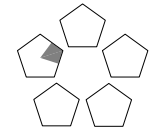
Broader applications

[[see published book for details]]

Resources

[[see published book for details]]

DE :: ARCHITECTURE ON PURPOSE



Principles

How do we know that what we do is 'on purpose'?

No matter what we do at work, it can't be effective unless it in some way supports the purpose of the enterprise. Which means that to do anything effective, we need first to know what that purpose *is*, and have some idea of how to get there.

[[see published book for details]]

Vision

[[see published book for details]]

Role

[[see published book for details]]

Mission

[[see published book for details]]

Goal

[[see published book for details]]

Common visioning errors

[[see published book for details]]

Procedure

[[see published book for details]]

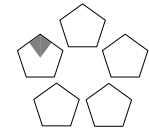
Broader applications

[[see published book for details]]

Resources

[[see published book for details]]

DL :: ARCHITECTURE IS A FEELING



Principles

What values are expressed in the architecture of the enterprise?

In a conventional IT-centric architecture, this kind of question has almost no meaning. In the Zachman framework, for example, the only reference to this is tucked away in the top right-hand corner of the 'Motivation' column. And values in the human sense aren't even mentioned in the TOGAF Architectural Design Method.

But as enterprise-architecture maturity increases, and is required to support more complex concerns such as organisational agility, adaptability and innovation, values necessarily come more to the fore.

[[see published book for details]]

Procedure

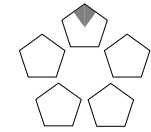
[[see published book for details]]

Broader applications

[[see published book for details]]

Resources

[[see published book for details]]



PL :: THE ARCHITECTURE TEAM

Principles

Who do we need for the architecture team?

The questions and concerns here are straightforward, though the answers will change considerably at differing levels of architecture maturity – see *MA :: Measuring maturity* (p.52). For the earliest stages there may not even be a team as such – more an occasional meeting of disparate specialists – and in later stages the architectural community-of-practice is likely again to be made up of dispersed ‘virtual teams’; but in the intermediate stages a tightly-coupled team is essential.

The make-up of the team is important at every stage, of course, but one of the most pressing concerns is leadership through the transitions *between* each of the maturity-levels. Twice now I’ve seen architectural development in a very large organisation come to a grinding halt because the then lead of the architecture team could not accept the need for a change in emphasis. In both cases the sticking-point was the transition from level-2 to level-3 – in other words, the shift from an inward-looking focus on technology to an outward-looking emphasis on business alignment for IT-architecture.

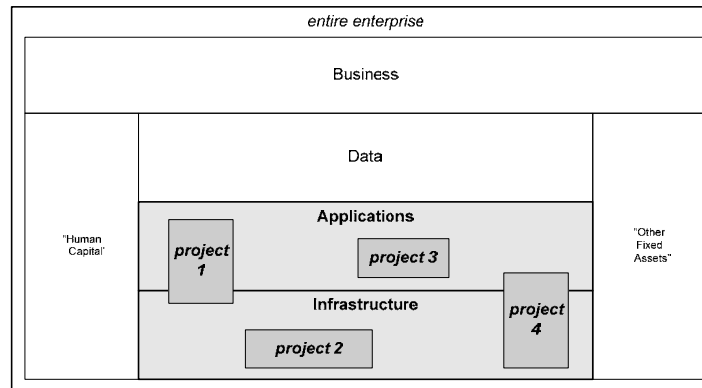
In one of the organisations, the lead was quietly promoted sideways to head a different IT team, and was replaced by someone with a stronger business-strategy background – after which the respect and usefulness of the enterprise-architecture grew by leaps and bounds.

But in the other organisation, unfortunately, the lead had gained the personal protection of a powerful patron at board level, with all-too-predictable results. Shielded from change, the architecture team retreated further and further into their known, ‘safe’ world of technology, even abandoning any interest in data-architecture as they spiralled down into angry irrelevance. Two years later, that organisation had lost almost all of its previous enterprise-architecture capability.

The individual and collective skill-sets will also change with each maturity-level, which we could summarise as follows.

In the initial *technical-architecture stage*, the main focus will be on reining in the cost and complexity of specific IT systems. It’s unlikely there would be any distinct enterprise-architecture team; instead, the architecture – such as it is – would be the responsibility of individual projects,

usually with a focus on either the application-layer or the infrastructure layer, though occasionally straddling the boundary between them.



Level 1: enterprise-architecture within projects

There's likely to be little or no focus on the data layer, only a limited awareness of the business layer, and no awareness at all of anything outside of an IT-oriented scope.

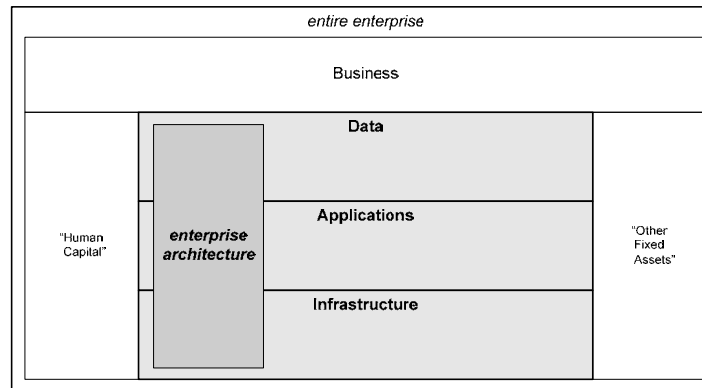
The skill-sets needed here are those with a straightforward technical focus: specific applications, specific network configurations, and so on.

Although in some projects there may be a requirement for specialist skills such as enterprise application integration, it's probable that any 'architecture' work will be done only by the project-lead, in coordinating the different technical specialists. There will at least be an awareness that the ability to construct that kind of integration does require a special type of skill.

For the next level, for *enterprise-wide IT-architecture*, a discrete 'enterprise architecture' team is created, usually as the result of issues identified by projects, or from repeated calls by business for some kind of cross-project coordination. In the early stages there may be only a handful of people in the team, but in the later stages there may well be dozens, especially in a larger organisation.

The focus is still strictly IT, but does now extend to include data-architecture. There will be a slightly stronger link with the business layer, particularly if coordination with process-changes is required for configuration of customer-relationship management, or business process

reengineering or the like, but beyond that there will still be little if any awareness of a world beyond IT.



Level 2: bridges data, applications and infrastructure

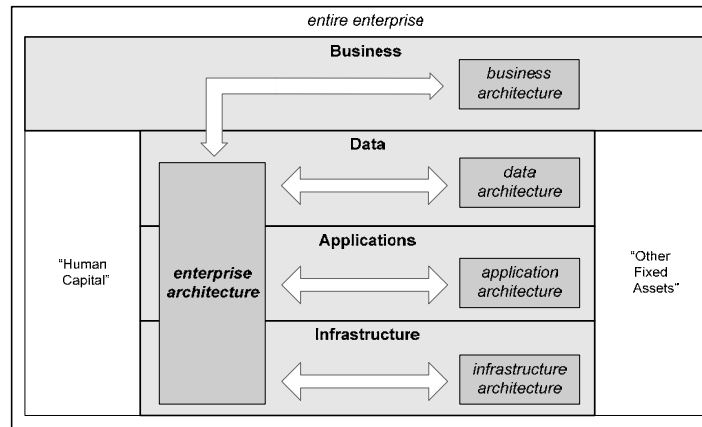
This period is characterised by construction of reference-models in the FEAF or Zachman mould, so the team will require not only a broad range of technical skills, but the ability to think in abstracts as well as the details of low-level implementation.

A classic danger here, often arising from confusions about the role of frameworks, is for the team to turn inward and wander off into an academic obsession with the purity of models for their own sake. So a key skill required of the team-lead here is the ability to keep a focus on the *usefulness* of the models for cross-project coordination. Without this, the team soon becomes irrelevant to business and even to the rest of IT, and the only useful enterprise-architecture development instead comes from 'transformation'-type projects – an embarrassing situation I've seen in practice in several large organisations.

Integration is usually provided by adherence to a standard framework – usually Zachman, at this stage, or an industry-specific framework such as eTOM for telecommunications.

The next stage, *IT-architecture with business-architecture*, is characterised by a shift to a stronger focus on the *application* of enterprise architecture – on methodology rather than only on models – and also on a closer alignment with business. The architecture team itself will often shrink radically, as distinct sub-groups for data-, application- and infrastructure architecture are spun off to provide specialist support to projects. There may also be crosslinks to new teams dealing with

architecture-type issues, such as service-oriented architecture, information-architecture, security-architecture and so on.



Level 3: distinct sub-architectures, stronger links with business

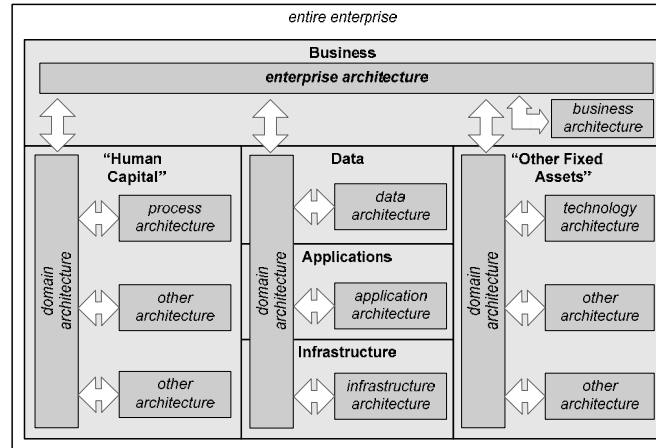
Freed from project-level architecture, the focus of the remaining enterprise-architecture team shifts to cross-project and cross-enterprise integration – searching for synergies and reducing possible system redundancies. There's stronger awareness of business drivers, and a whole new area loosely labelled as 'business architecture' – even though the term is often misused to mean 'anything not-IT'.

The required skill-sets also change radically: team-members now need to be generalists bridging *across* all the specialist domains, and able to hold the big-picture view in their heads whilst assessing project-level detail – see *KA :: Architecture as a way of thinking* (p.37) and *TR :: The practice of architecture* (p.42). Interpersonal skills also come to the fore, in promoting a more disciplined approach to governance, and in cajoling often-recalcitrant project-leads to comply with the selected architecture principles – though some of this load should be taken over by the programme management office as governance-maturity is further developed across the enterprise.

The whole team is responsible for cross-project integration, guided by a formal methodology such as FEAF or TOGAF.

At the true *enterprise-architecture* level, the role moves 'upward' to become responsible for architecture across the enterprise as a whole – and not merely for the IT subset. (A hint: at this

level, if a job-specification includes a reference to any specific technology, such as .NET or SAP or whatever, it's no longer *enterprise* architecture – it's an aspect of domain-level enterprise-IT-architecture. It may seem a subtle difference, but it's extremely important here.)



Level 4: coordinates architecture across whole of enterprise

The previous IT-oriented ‘enterprise architecture’ becomes one of several vertical *domain architectures*, each coordinating the specialist sub-groups working across a cluster of organisational ‘silos’, partitioned by business function, geographical location or the like.

Where the previous team would report to the CIO or CTO (and probably still does), this new team will usually be attached to a cross-enterprise group such as strategy or the programme management office, or even report direct to the CEO.

For the first time, the focus is truly enterprise-wide. (In the diagrams here I’ve used the FEAF labels “Human Capital” and “Other Fixed Assets” to indicate these previously-‘invisible’ aspects of the organisation.) The emphasis is on a much stronger integration with business strategy, and on cross-domain concerns such as increased agility and innovation, reduced time-to-market for new products and services, and improved potential for partnerships to extend the effective boundaries of the enterprise.

Members of this small team need to be strong generalists – see *PN :: The role of the generalist* (p.30) – who are able to listen well, to learn fast, to synthesise ideas quickly in visual or verbal form, and

able to build easy relationships with anyone at any level in the organisation and beyond. Although they should usually have a background in a relevant technical or business discipline, they are more likely to call on the team's network of specialist advisers, from within the domain-architecture sub-groups or elsewhere.

Although often associated with the business-strategy group, the roles of the teams are different: strategy defines the desired direction, whilst enterprise-architecture identifies the connections that make the strategy possible in practice.

Procedure

Purpose

Identify the make-up of the enterprise-architecture team, the required collective and individual skill-sets, and recommended collective and individual performance-criteria.

People

Selected senior managers, enterprise architects, lead trainers, HR specialists.

Preparation

Standard business-analysis tools: whiteboard, meeting-space, pen and paper, time, and access to the required people.

CVs and broader interests of existing members and candidates for the architecture team.

Assess the maturity-level of the present architecture before you start – see *MA :: Measuring maturity* (p.52).

Process

Use literature search, and standard business-analysis interviews and group facilitation with key stakeholders such as managers of 'customer' groups, to elicit a list of skill-sets and personal attributes required collectively by members of the architecture team, and by each individual member of the team.

For the earlier maturity-levels, the emphasis will be more on technology- or domain-specific skills:

- *technology/infrastructure architecture*: development of technology principles, standards and patterns that can be shared across multiple systems and solutions
- *applications architecture*: aligning the solutions portfolio with the business and data architectures, and linking applications together
- *data/information architecture*: modelling, classification, integration and management of data, metadata and information, and the business-rules linking them and guiding their transformations
- *business architecture*: modelling and management of processes, workflows, material-flows and the like, and strategy and change-management to guide functional change

Although relevant at all stages, the core *architectural* skills come to the fore at later maturity-levels:

- communicator, change-agent, and creator of social-networks
- modeller and visual thinker
- strategist with foresight disciplines
- fast learner
- principled pragmatist
- consultant and 'troubleshooter'
- 'big picture' thinker

Using the identified maturity-level and individual CVs as a guide, assess the current architecture-team for gap-analysis to identify:

- changes to team make-up: additional members needed, sub-teams split-off, and so on
- additional skills: training required, conference-attendance etcetera
- changes to reporting and line-of-authority: to project-group, to CTO/CIO, to programme management office, to strategy/CEO, etcetera

Develop appropriate strategy and management for any required changes. Where necessary, document these changes in the architecture governance charter.

Performance (artefacts and outcomes)

List of required skill-sets; content for enterprise-architecture job-specifications and performance criteria; gap-analysis and change-management strategy for amendments to enterprise-architecture team; content for enterprise-architecture governance-charter.

Broader applications

This perspective is *People / Elegant* – a recursive emphasis on people *as* people, in all their complexity and difference; on human-factors, such as ergonomics and self-adaptation; and also on simplicity, clarity, the *effortlessness* of elegant design and operation when these all act ‘on purpose’.

In this section we’ve focussed on how these issues come out in the skills and attributes needed for the architecture team. But the same principles and drivers apply to the skills and attributes needed elsewhere:

- what is the purpose? – see *DE :: Architecture on purpose* (p.16)
- what is the scope? – see *DN :: Architecture of the enterprise* (p.14)
- what is the maturity-level? – see *MA :: Measuring maturity* (p.52)

We also need to pay especial attention to the human complexities of politics in the enterprise – see *PA :: The politics of purpose* (p.28) – and the subtle problems of misperceptions of power – see *PR :: A problem of power* (p.29) – because all of these will impact every aspect of the enterprise. In each case, though, clarity on purpose is likely to provide the guiding star that’s most needed to navigate through the dangerous shoals and stormy waters.

The same themes – or similar themes – also apply in other areas such as customer-relationship management, customer relationship training, work/life balance, product-personalisation and many others. All of these are about the links between people and work, or between people and the enterprise in general. There are plenty of tools and techniques to address these issues: use them.

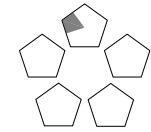
Resources

🏠 Zachman framework: www.zifa.org

🏠 FEAF: www.cio.gov

🏠 TOGAF: see www.opengroup.org/architecture/togaf8-doc/arch/toc.html

🏠 Sally Bean, "The elusive enterprise architect", IT Adviser, Issue 43, May/June 2006;
www.nccmembership.co.uk/pooled/articles/BF_WEBART/view.asp?Q=BF_WEBART_205593



PA ::THE POLITICS OF PURPOSE

Principles

Whose work will be affected by any architectural strategy? In other words, who are the stakeholders in the architecture? And what will be their concerns with the architecture?

This is another one of those sections where in principle the issues should be simple and straightforward, yet in practice are usually anything *but* simple – because here is where we meet head-on with the tangled politics of change.

[[see published book for details]]

Procedure

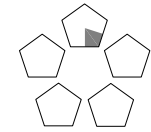
[[see published book for details]]

Broader applications

[[see published book for details]]

Resources

[[see published book for details]]



PR :: A PROBLEM OF POWER

Principles

What is the organisation's ability to do work?

This is another question that seems simple, and to which the answers should also be simple. But they're not. At all. In fact, it's the core of many of the issues that an enterprise-wide architecture aims to resolve.

[[see published book for details]]

Procedure

[[see published book for details]]

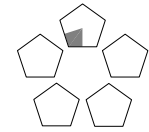
Broader applications

[[see published book for details]]

Resources

[[see published book for details]]

PN :: THE ROLE OF THE GENERALIST



Principles

Who are your generalists – the people who link *across* the different business domains? What support do they have in this role? And what support do they need?

This is another of these somewhat political aspects of enterprise-architecture. But at least it's less fraught than most we've seen so far, because the politics arise not from personalities but from the nature of organisations and their structures - which does take *some* of the heat out of the issue.

[[see published book for details]]

Procedure

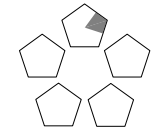
[[see published book for details]]

Broader applications

[[see published book for details]]

Resources

[[see published book for details]]



PE ::WHAT'S THE STORY?

Principles

From where or whom does the organisation's knowledge arise? Who identifies and creates business meaning? And how is this 'tacit knowledge' communicated, managed, refreshed, sustained?

From a knowledge perspective, many current 'enterprise architecture' frameworks such as Zachman, FEAF and TOGAF can be intensely frustrating, because they promote the delusion that knowledge is synonymous with IT. It isn't: in fact we could almost describe this as a kind of laziness, because although managing IT-based knowledge is rarely easy, managing the non-IT-based knowledge is *hard*.

[[see published book for details]]

Procedure

[[see published book for details]]

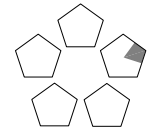
Broader applications

[[see published book for details]]

Resources

[[see published book for details]]

KE :: DIMENSIONS OF ARCHITECTURE



Principles

What information do you need, to identify and improve the effectiveness of the enterprise? What framework would you use to model this information?

[[see published book for details]]

Procedure

[[see published book for details]]

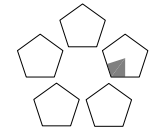
Broader applications

[[see published book for details]]

Resources

[[see published book for details]]

KN ::AN EMPHASIS ON EFFECTIVENESS



Principles

What do we need to make the enterprise more effective?

If we ask business-people what they expect from enterprise architecture, the most common answers are “increase efficiency” and “reduce costs”. But it’s quite easy to increase efficiency in one area by reducing the efficiency even more in others. And it’s easy to reduce short-term costs, by destroying long-term capability. What matters most is not efficiency, but overall *effectiveness*.

For almost a century now, Frederick Taylor’s theories of ‘scientific management’ have held sway, with their insistent – some would say obsessive – focus on efficiency and control. But in practice, as W Edwards Deming and others have demonstrated so well, Taylorism only works well for enterprises with stable products, stable processes, stable markets and rigidly-prescribed work-roles – which applies to very few of today’s business contexts. For the rest, a focus on efficiency alone is a recipe for expensive disaster.

There’s a well-known example from when a large multi-national bought the consulting arm of another very large firm.

The price had a terrifying number of zeroes after the currency-sign, but it had seemed such a great idea: ‘a marriage made in heaven’. The consultants added great cachet to the company, and just the possibility of access to that dream-list of clients sent their salesmen salivating.

But along came the company’s bean-counters, who wanted “a few changes” in the name of efficiency. So gone were the consultants’ coffee-machines; gone was any idea of personal workspace, replaced by ‘cube-farm’ hot-desks; gone even were any private rooms for meetings with clients. Oh, and performance was now to be measured solely on the number of boxes they sold – when none of them had ever sold anything in their working life. It was *not* a happy time...

I happened to be visiting on the day everything went pear-shaped. Within weeks, *all* the consultants had left the new company, taking their clients with them – and there was nothing the company could do about it. All that remained from their huge investment was a name – and a few months later even that had faded into history.

Very efficient, yes. But not very effective.

What we need is not just improved efficiency, but improvements in overall *effectiveness*. And no, it's *not* easy. But we can make it easier by being clear about what the components of effectiveness are. They map closely to the dimensions of the framework we're using here: Purpose, People, Preparation, Process, Performance:

- **Efficient** – optimise use of resources, and minimise waste. This comes from careful observation and re-planning, and fits well with the *Preparation* dimension. The observation and re-planning may be done by anyone appropriate – as in *kaizen* continuous improvement in TQM, for example.
- **Reliable** – timely, predictable, consistent, self-correcting. As might be expected, this corresponds with the *Process* dimension, especially in the physical domain.
- **Elegant** – clarity, simplicity, self-adjusting for human factors. Although this matches the *People* dimension, this is 'elegance' as much in the scientific sense – in other words clarity and the like, not for their own sake, but because it simplifies re-use, maintenance, adaptability, agility and pro-active response.
- **Appropriate** – supports and optimises support for business purpose. The *Purpose* dimension is essential to effectiveness, providing guidelines to allow activities to be 'on purpose'.
- **Integrated** – creates, supports and optimises synergy across all systems. This relates to the *Performance* dimension, because overall performance arises from a complex of outcomes which depend in turn on integration across all the functions of the enterprise.

Some applications of this checklist include:

- the overall 5Ps enterprise-architecture framework, to assess effectiveness across the entire enterprise and its interfaces with 'outside' agencies and organisations;
- the SCORE checklist in strategic assessment, to verify impact of strategy on overall effectiveness – see *TA :: What's the SCORE?* (p.48);
- structured feedback into the architecture, to improve the effectiveness of the architecture – see *ME :: Closing the loop* (p.50);
- implementations of Balanced Scorecard and other integrative metrics – see *MR :: Real-time scoreboards* (p.49).

Procedure

Purpose

Establish frameworks for assessment, review and measurement of cross-functional effectiveness.

People

Enterprise architects, strategists, process architects, knowledge managers, senior operations managers, lead trainers.

Preparation

Standard business-analysis tools: whiteboard, meeting-space, pen and paper, time, and access to the required people.

Research on internet and other sources, for methodologies on improving all aspects of effectiveness.

Assessments of existing capabilities and facilities for measuring and enhancing effectiveness at function and cross-function levels.

Process

Review existing methods for assessing effectiveness against the checklist above: efficient, reliable, elegant, appropriate, integrated.

Identify and document any changes required to improve effectiveness across the enterprise, and monitoring of effectiveness.

Outline change-requirements to amend performance metrics and training procedures accordingly.

Performance (artefacts and outcomes)

Metrics and content for cross-functional scorecards; content for cross-functional training and assessment procedures.

Broader applications

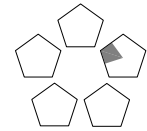
This perspective is *Preparation / Integrated* – an emphasis on how an awareness of the whole can be used to drive process improvement. Its direct counterpart is *Performance / Efficient*, on the ways in which shared knowledge can be used to pull the enterprise together – see *ME :: Closing the loop* (p.50).

The effectiveness-checklist above, and, in turn, this entire 5Ps framework, should apply not just to enterprise-architecture, but to every aspect of the organisation. All I can suggest here is: try it. See what happens. Keep applying it recursively, iteratively, as a means to highlight potential for overall improvement.

Resources

🏠 Tetradian SEMPER diagnostic: www.tetradian.com/semper

KA :: ARCHITECTURE AS A WAY OF THINKING



Principles

What principles and frameworks will you use to guide your own thinking about the enterprise-architecture?

[[see published book for details]]

Procedure

[[see published book for details]]

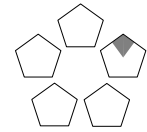
Broader applications

[[see published book for details]]

Resources

[[see published book for details]]

KL ::A QUESTION OF RESPONSIBILITY



Principles

Who's responsible for each of the enterprise's knowledge-assets? What are these assets? How are the responsibilities identified, supported, maintained and transferred?

[[see published book for details]]

[[see published book for details]]

Procedure

[[see published book for details]]

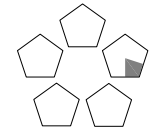
Broader applications

[[see published book for details]]

Resources

[[see published book for details]]

KR :: THE CENTRALITY OF SERVICES



Principles

What services are described in the architecture of the enterprise? What interfaces do these services expose? How are these interfaces published? Who or what are the consumers of these services? And how do service-levels change with varying implementations of each service?

[[see published book for details]]

Procedure

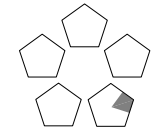
[[see published book for details]]

Broader applications

[[see published book for details]]

Resources

[[see published book for details]]



TE :: REQUIREMENTS FOR AGILITY

Principles

What requirements and other information-items are needed, to drive enterprise agility? How are they collected, collated, monitored, maintained, reviewed, re-used?

[[see published book for details]]

Procedure

[[see published book for details]]

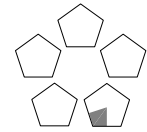
Broader applications

[[see published book for details]]

Resources

[[see published book for details]]

TN :: MANAGING SERVICES



Principles

In defining a service-oriented architecture, how will those services be managed? How does service-management intersect with your enterprise-architecture? In what ways can you leverage service-management to enhance integration across the whole architecture?

[[see published book for details]]

Procedure

[[see published book for details]]

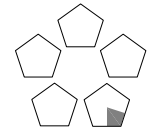
Broader applications

[[see published book for details]]

Resources

[[see published book for details]]

TR :: THE PRACTICE OF ARCHITECTURE



Principles

What will you *do* to define and implement the enterprise architecture? What methodology should you use to guide your architecture practice?

The keyword here is *practice*. This isn't about the guiding ideas, or the mindset, or whatever – for which see *DE :: Architecture on purpose* (p.16) or *KE :: Dimensions of architecture* (p.32) – but about what we *do*, and how we do it. So the three main concerns here are methodology, toolsets, and governance of compliance.

[[see published book for details]]

Procedure

[[see published book for details]]

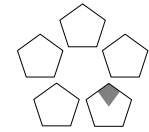
Broader applications

[[see published book for details]]

Resources

[[see published book for details]]

TL :: THE ART OF INTEGRATION



Principles

What can you use to unify every aspect of the enterprise? What theme would drive transformation, and lift performance to a whole new level?

This is one of the best ways to boost overall performance, for everyone – *if* we get it right. It's also almost independent of architecture maturity, though it works best at higher maturity-levels. The hard part is in identifying where to start: an inappropriate theme goes nowhere – though rarely does any damage – but the right one takes off in ways that are truly as spectacular.

A routine board-level presentation, for a routine kind of project, reviewing procedures and systems at a recruitment company for compliance to new privacy legislation. About halfway through, we came to a slide about relationships with candidates, clients and others, and I described the company's role as that of a 'trusted intermediary'.

Sudden silence. You could have heard a pin drop.

"Can you say that again, please?" That was the managing director.

"I'm saying your real role is as a trusted intermediary. It's not just one way, obtaining candidates for clients, like some kind of cannon-fodder. In many cases, your candidates *become* your clients, and they also find new clients for you. So the real issue here isn't privacy, but *trust*. If we review all the processes and so on through that lens of trust, always looking for ways to enhance that trust, what you'll have is a business model that extends itself automatically. And one that gets its income from every direction, because, for every one of your stakeholders, you act as their trusted intermediary."

Again that deep, disturbing silence, finally broken once more by the managing director: "D'you reckon we could patent that?"

This is a corollary of systems-theory, particularly the concept of reflexion – see 'Systems thinking' in *KA :: Architecture as a way of thinking* (p.**Error! Bookmark not defined.**). In a systems-view, every aspect of the enterprise is linked to and part of every other, in much the same that all the organs in a living body are intimately connected with and dependent on each other. So to improve overall performance, it doesn't really matter where we start, as long as we start *somewhere*.

In principle we could use any theme at all. In reality, most don't get very far: they quietly fade away to nothing. What we're after is something *self-propagating* – the 'snowball effect' from that other systems-principle of resonance. To do this, the theme must:

- be *universal* – applies to every aspect of the enterprise
- be *emotive* in some way
- be *directly applicable* in everyday, personal work
- be applicable *iteratively*, permitting continuous improvement
- be *measurable*, such that improvements can be tracked

Probably the best example is a focus on quality. W Edwards Deming used it in a number of variations – *kaizen*, *kanban* and so on – to transform Japanese industry. Jack Welch used it to create Six Sigma at General Electric. It engages everyone; it's real, measurable, tangible; it works. But it isn't the only option: others I've used include:

- client trust in privacy – as in the story above
- client trust in delivery – reliability of logistics
- corrective action – process improvement
- 'single source of truth' – data quality and data management
- waste-reduction – time, money, materials, other resources

What works best is something that matches the needs and mood of the enterprise. Jack Welch selected several such themes in his tenure at General Electric, but only ever *one* theme at a time, and each for only three to five years. Sometimes a theme presents itself as 'the obvious thing to do'; for others, it can be helpful to conduct some kind of narrative-inquiry such as Open Space – see *PE :: What's the story?* (p.31) – to elicit an appropriate theme from within the collective tacit knowledge of the enterprise.

Another key advantage of this type of integration strategy is that it tends to pull the power-dynamics toward the functional end of the spectrum – see *PR :: A problem of power* (p.29). Part of this is from the natural enthusiasm of the 'local champions' who take on the strategy as their own – see *PA :: The politics of purpose* (p.28). But it's also because the theme engages people personally and emotively in new ways of looking at their work, and makes the work inherently feel more worthwhile – the same driver for constructive change as in *DL :: Architecture is a feeling* (p.18).

Procedure

Purpose

Identify and establish frameworks to use one or more unifying 'themes' for whole-of-organisation integration.

People

Selected senior managers, strategists, enterprise architects, change management specialists, others as appropriate (especially any local 'champions' for a selected unifying theme).

Preparation

Standard business-analysis tools: whiteboard, meeting-space, pen and paper, time, and access to the required people.

If appropriate, conduct narrative-enquiry – see *PE :: What's the story?* (p.31) – to elicit candidate themes.

Process

Together with all members of the team above, brainstorm and review potential themes for whole-of-organisation integration.

Verify each candidate against the checklist above: universal, emotive, personal, applicable, iterative and measurable. Identify metrics to monitor improved performance in an overt, easily-understandable form – see *MR :: Real-time scoreboards* (p.49).

With the assistance of selected local 'champions', pilot-test potential changes to procedures and feedback – see *ME :: Closing the loop* (p.50) – using the complexity principle of 'probe / sense / respond' – see 'Managing complexity' in *KA :: Architecture as a way of thinking* (p.**Error! Bookmark not defined.**) – and providing appropriate support for knowledge-sharing – see *PE :: What's the story?* (p.31) and *KL :: A question of responsibility* (p.38).

If a selected theme seems to self-propagate appropriately, roll out into broader usage, ideally to the scale of the entire enterprise.

Monitor and repeat as required.

Performance (artefacts and outcomes)

Strategy-level and tactics-level plans for roll-out of selected unifying theme, including content for change-management, risk/opportunity management and governance.

Broader applications

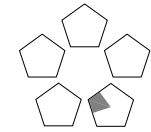
This perspective is *Process / Elegant* – an emphasis on the use of emotive ‘themes’ and other tactics to drive process-improvement. Its direct counterpart is *People / Reliable*, about tactics to make the actions of and interactions between people easier and more productive – see *PR :: A problem of power* (p.29).

The description and procedure are not only for any aspect of the enterprise, but ideally should always apply to the *whole* enterprise. That’s how it works best, not only because it improves integration anyway, but because by engaging *everyone* in the emerging conversation, the maximum possible range of ideas and experiences can be brought to bear on any issue or problem. Other useful ‘themes’ here can include ‘post-compliance’ TQM such as quality-circles; an emphasis on occupational health and safety, environment; ergonomics, and personalisation, both within the organisation – such as integrated performance-support systems – and outside – such as ‘market of one’ customisation.

The guiding principle for choice of ‘theme’ is as described above: “what works best is something that matches the needs and mood of the enterprise at the time”. It’s essential, then, to listen out for hints and signals about the changing nature of the enterprise, to catch the right moment to inject an appropriate ‘theme’. As above, large-group interventions such as Open Space are probably the most valuable tactics to use here; but in the absence of those, or a lack of maturity to enable such tactics to be safely used, monitor the mood via some other means such as the SEMPER whole-of-context diagnostic – see *ML :: People and performance* (p.51) – and use the results to identify an appropriate intervention.

Resources

- 📖 Jack Welch at GE: see Robert Slater, *Jack Welch & The G.E. Way: Management Insights and Leadership Secrets of the Legendary CEO*, McGraw-Hill, 1998, ISBN 0070581045
- 📖 Peter M Sengé et al., *The Fifth Discipline Fieldbook: Strategies and tools for building a learning organization*, Currency (1994), ISBN-13 978-0385472562.
- 🌐 Open Space: www.openspaceworld.org
- 🌐 TQM: see www.managementhelp.org/quality/tqm/tqm.htm



TA ::WHAT'S THE SCORE?

Principles

How can you ensure that each strategy supports every aspect of the enterprise?

This is another straightforward section that can be applied at any maturity-level. What we need here is some means to assess the probable impact of a strategy or tactic, not only on the immediate area of interest, but on the *whole* enterprise.

[[see published book for details]]

Procedure

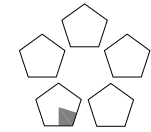
[[see published book for details]]

Broader applications

[[see published book for details]]

Resources

[[see published book for details]]



MR :: REAL-TIME SCOREBOARDS

Principles

In what ways can the architecture guide the monitoring of real-time performance? How do you ensure that the most relevant, meaningful and accurate metrics are used? How do these metrics reflect the performance of the *whole* of the enterprise?

[[see published book for details]]

Procedure

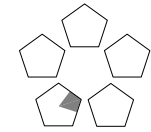
[[see published book for details]]

Broader applications

[[see published book for details]]

Resources

[[see published book for details]]



ME :: CLOSING THE LOOP

Principles

How do you validate the architecture? What processes would you use to obtain and incorporate feedback from the architecture's stakeholders? How can you use the architecture itself to drive a process of continuous learning and growth?

[[see published book for details]]

Procedure

[[see published book for details]]

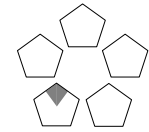
Broader applications

[[see published book for details]]

Resources

[[see published book for details]]

ML :: PEOPLE AND PERFORMANCE



Principles

What is the organisation's ability to do work? How would you measure and monitor this?

Financial figures and the like are useful, of course, but they're rearward-looking 'lag-indicators' that tell us where we've been – whereas what we need are 'lead-indicators' that tell us whether we're on track to where we want to go. So what we're looking for here are metrics that point to *future* performance – especially about people and the collective 'ability to do work'.

[[see published book for details]]

Procedure

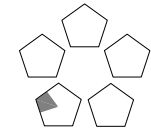
[[see published book for details]]

Broader applications

[[see published book for details]]

Resources

[[see published book for details]]



MA :: MEASURING MATURITY

Principles

What is the current maturity of your enterprise-architecture? What is its most appropriate scope? And what next steps do you need to take to extend that maturity and scope?

Maturity-metrics do provide useful indicators as to the probable value of the architecture to the enterprise – see *MN :: Monitoring integration* (p.53). But their real purpose is much simpler, namely to guide what to develop next:

- changes to the team – see *PL :: The architecture team* (p.19)
- changes to scope – see *DN :: Architecture of the enterprise* (p.14)
- changes to architecture practice – see *TR :: The practice of architecture* (p.42)
- changes to the guiding ideas and perspectives – see *KA :: Architecture as a way of thinking* (p.37), *KE :: Dimensions of architecture* (p.32) and *DL :: Architecture is a feeling* (p.18)

[[see published book for details]]

Procedure

[[see published book for details]]

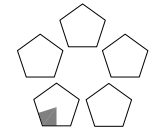
Broader applications

[[see published book for details]]

Resources

[[see published book for details]]

MN :: MONITORING INTEGRATION



Principles

How can you demonstrate the value of enterprise-architecture? What is the architecture worth to the enterprise as a whole?

This one's always going to be hard. No surprises in that, because architecture isn't a production environment – it supports production, but doesn't produce anything itself – so will always be seen first as a cost. As with risk-management, most of the value gained is in terms of what *didn't* happen – which is much harder to prove and to quantify *because* it didn't happen.

[[see published book for details]]

Project-level IT-architecture

[[see published book for details]]

Enterprise-wide IT architecture

[[see published book for details]]

Enterprise IT-architecture with business-architecture

[[see published book for details]]

Whole-of-enterprise architecture

[[see published book for details]]

Procedure

[[see published book for details]]

Broader applications

[[see published book for details]]

Resources

[[see published book for details]]

GLOSSARY

This summarises some of the terms and acronyms we've come across in the book.

[[see published book for details]]