An Overview to Enterprise Architecture

Jorge Luis Rodriguez

June 2024

"If you think good architecture is expensive, try bad architecture"
Brian Foote and Joseph Yoder

"Architecture is always political"
Richard Rogers

"You've got to think about big things while you're doing small things, so that all the small things go in the right direction"
Alvin Toffler

Table of Contents

Table of Figures

Abstract

A fundamental difference exists between an enterprise operator and an enterprise designer. A manager runs an organization just as a pilot runs an airplane. Success of a pilot depends on an aircraft designer who created a successful airplane.... Who designed the corporation that a manager runs? (Rao, Reedy, & Bellman, 2019).

It is true that there are many deep differences among the roles and responsibilities between the airplane pilot and the enterprise manager, but (Rao, Reedy, & Bellman, 2019), is just pretending to illustrate the importance of the organizational arrangement of any enterprise.

Enterprise Architecture, a relatively new discipline (around since 1980's), came as (Group, 2018) states, with the purpose of optimizing an often-fragmented legacy of processes into an integrated environment to better respond to changes, while supporting the delivery of the business strategy.

Many companies are loosely adopting TOGAF, to achieve effective and efficient operations, to guide the digital transformation and IT operations, to gain better returns on investments whilst minimizing the risks, and to improve procurement.

But as (Rother, 2009) explains, there is an unspoken frustration because of a gap between desired results and what really happens. Targets are set, but they are not reached. Change does not take place as planned.

This paper is attempting to address some of the Information Technology – Enterprise Architecture difficulties faced by many companies and offer some suggestions on how to overcome them.

Introduction

In 1986, (Brooks Jr., 2010) wrote:

"Of all the monsters who fill the nightmares of our folklore, none terrify more than werewolves, because they transform unexpectedly from the familiar into horrors. For these, we seek bullets of silver that can magically lay them to rest".

"The familiar software project has something of this character (at least as seen by the nontechnical manager), usually innocent and straightforward, but capable of becoming a monster of missed schedules, blown budgets, and flawed products. So, we hear desperate cries for a silver bullet, something to make software costs drop as rapidly as computer hardware costs do".

"But, as we look to the horizon of a decade hence, we see no silver bullet. There is no single development, in either technology or management technique, which by itself promises even one order of magnitude improvement in productivity, in reliability, in simplicity".

In fact, as (Wierda, Chess and the art of Enterprise Architecture, 2015) points, using an ESB, a BRE, SOA, Cloud, and all the others does not by itself provide stability, robustness, and good and future-proof systems.

"Without some of them, we might indeed end up with systems dying in a gridlock of hard dependencies, true, but the price we pay is an increase of loosely coupled spaghetti-like dependencies. And IT is not only complex, but also fundamentally brittle because, as a digital environment, it lacks the flexibility of human type compensating behavior. These days — thanks to networking and especially the internet — these dependencies range across all kinds of levels: software, application servers, networks and infrastructure, and even

organizations. In part, it is managing this complexity of dependencies that the field of enterprise architecture was originally invented for". (Wierda, Chess and the art of Enterprise Architecture, 2015)

Most enterprises possess a wide variety of interconnected components using multiple operating systems, programming languages, databases and technologies; among them: Mainframe, Windows, Linux, C, C++, .NET, C#, Prolog, Java EE, Cloud Services (SaaS, PaaS, IaaS), OpenShift, Stackato, IBM WAS, MongoDB, Redis, DB2, MS SQL Server, MySQL, Apache Tomcat, Kafka, Jenkins, Git, Docker, Power BI, Qlik, etc.

Would enterprise architecture be able to resolve this kind of problem of essential complexity or would it become another silver bullet itself? Should enterprise architects maintain the position that the essential complexity is solvable at all?

Companies initially saw the problems, to be 'alignment' of various sorts, and the solution was Enterprise Architecture, even though it was not initially called that way.

"Various Enterprise Architecture methodologies have arisen to manage/produce this alignment. With names like Zachman, DYA, DoDAF, MODAF, TOGAF, IAF, FEAF, NAF, NORA, ESAAF, GEA, TEAF, LEAD, AGATE, Praxeme, TRAK, Dragon1, SABSA, they now are an integral part of large IT-oriented change management". (Wierda, Chess and the art of Enterprise Architecture, 2015)

I believe that enterprise architecture has not delivered all what was assumed it would, but several companies, with all their increasing complexities; very large IT systems are continually working, improving and modifying. They are not only doing things wrong, they are also doing things right.

Complexity

"Complexity studies should thus be seen not as aiming at a new "synthetic theory" of complexity of any kind, but as a cross disciplinary field of research and meeting place for dialogue between specialized groups of people such as biologists, physicists, philosophers, mathematicians, computer scientists, and, last but not least, science writers (with a background in science or journalism or both) who have contributed to popularize the field for a wider public and perhaps facilitated the meeting of experts from the specialized areas". (Emmeche, 1997)

Complexity, when used in connection with science, may have few general meanings as mentioned by (Emmeche, 1997):

- Descriptive complexity: applied to situations when several different methods are needed to describe a phenomenon in a reasonably complete way.
- Ontological complexity: when something is organized as a system of many non-identical components who themselves have systems-like properties (such as being further decomposable), and whose mutual interactions bring forth a kind of collective behavior which is different from the behavior of the parts.
- Complex dynamic systems or also called adaptive systems: endeavors to investigate self-organizing systems, co-operative behavior of agents, and non-linear dynamical systems creating emergent properties during their time evolution. This is causing growing interest from economics and social science.
- Complexity from the philosophy of science viewpoint: role of causality, interlevel relations and predictions in science.
- Complexity in social sciences: dealing with complicated social systems, their differentiation and segmentation, and with the various decision-making processes in these systems that constantly rely on incomplete information. In the theory of Niklas Luhmann, complexity reduction is the phenomenon

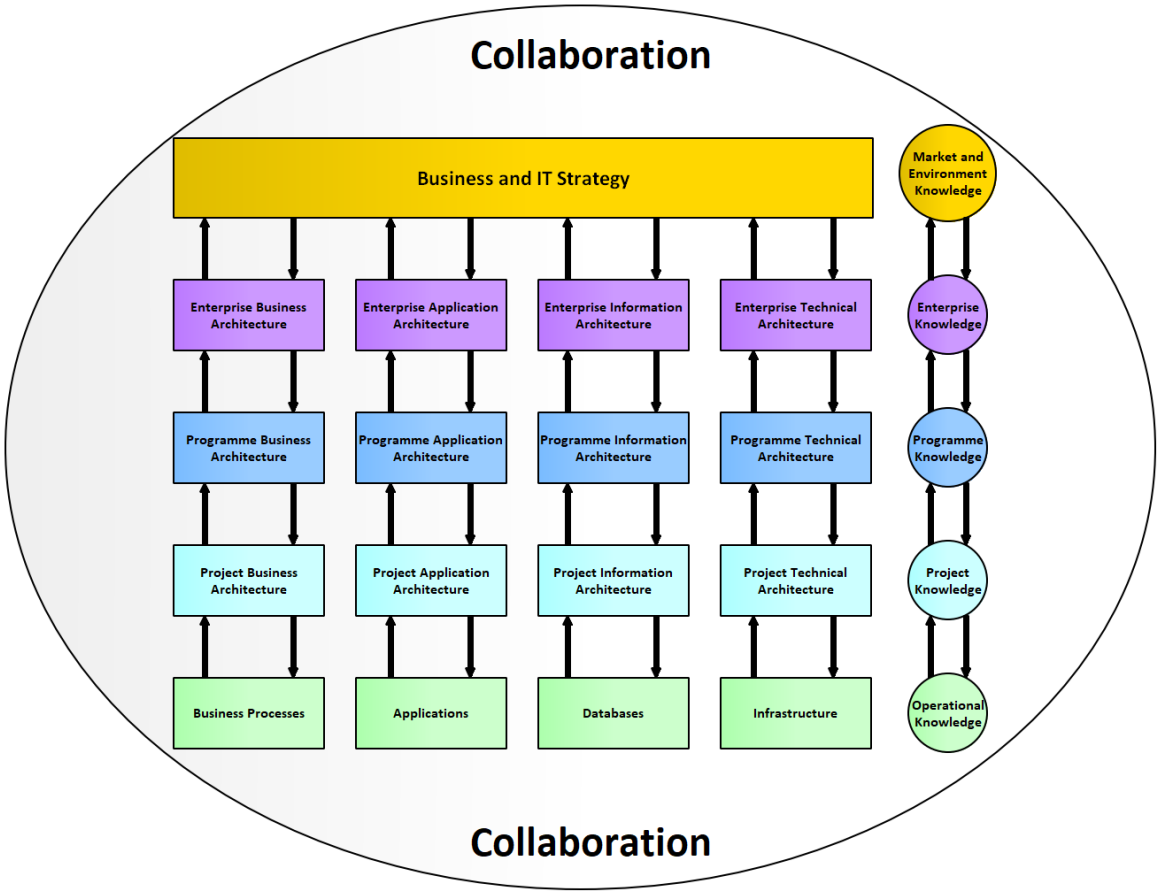
that social systems are exposed to much greater 'information pressure' than they can handle in real time by rational methods.

Enterprise Architecture usually has been compared to city planning or some other elements of the physical architecture world, but this comparison fails to notice the high level of dynamic complexity and unpredictability that EA must deal with.

(Wierda, Chess and the art of Enterprise Architecture, 2015) considers that a better analogy for EA is the game of chess, because a grand master concentrates on making good moves (tactics) while keeping a vision of them (strategy). But still this likeness is short to the reality of the EA complexity, which is many orders higher. Chess has only six different pieces with specific and defined movement capabilities (maximum of 32 pieces), played between two opponents on opposite sides of a 64 squares board of alternating colors, who take turns to move their pieces following strict logical rules. Enterprise Architecture needs to fight unpredictability and unknowability, where many players are simultaneously making changes on some giant board without taking turns.

The logical question would be: how should enterprises handle their inherent complexity and unpredictability? As history has shown, human beings have fought complexity with their own complexity: collaboration. Effective collaboration.

Figure 1: Fighting Complexity



Abstraction, as used in Computer Science for managing complexity of computer systems, works by establishing levels of complexity on which a user/person interacts with the system, suppressing the more complex details

below the current level. Unfortunately, when working on enterprise landscapes, chaos may look very clean when it is abstracted enough, creating false and simplistic impressions that could induce erroneous decisions.

“The behavior of classical enterprise architecture is also reminiscent of that old joke about the drunk looking for his lost keys under a streetlight. When asked by the passing cop what he is doing, he says: “I’m looking for my keys". The cop asks: "Did you lose them here?". "No", says the drunk, "I lost them back there". "So why are you looking for them here, then?" asks the policeman? "Well," says the drunk, "here is the light". (Wierda, Chess and the art of Enterprise Architecture, 2015)

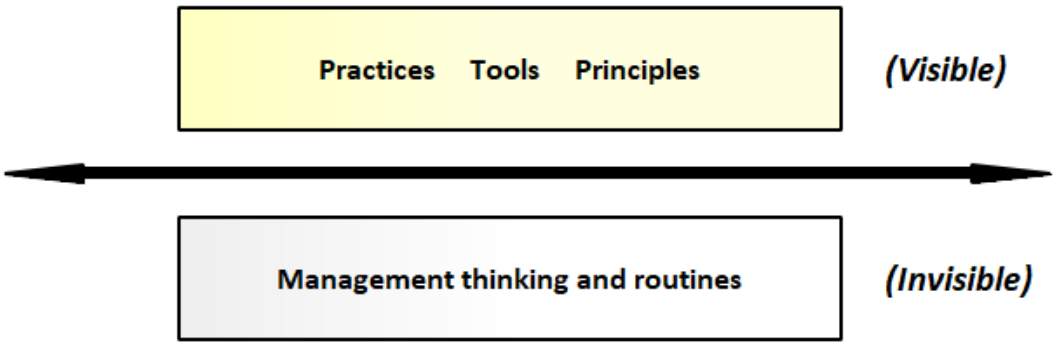
Principles, Standards, and Guidelines

“Tools, rules, techniques are necessary, but not enough to move an organization toward a culture of continuous improvement. It is indeed the daily behavior of the people inside the framework of the organization that defines the culture”. (Rother, 2009)

(Group, 2018) recommends the adherence to principles, standards and requirements, which should be monitored by the Architecture Governance.

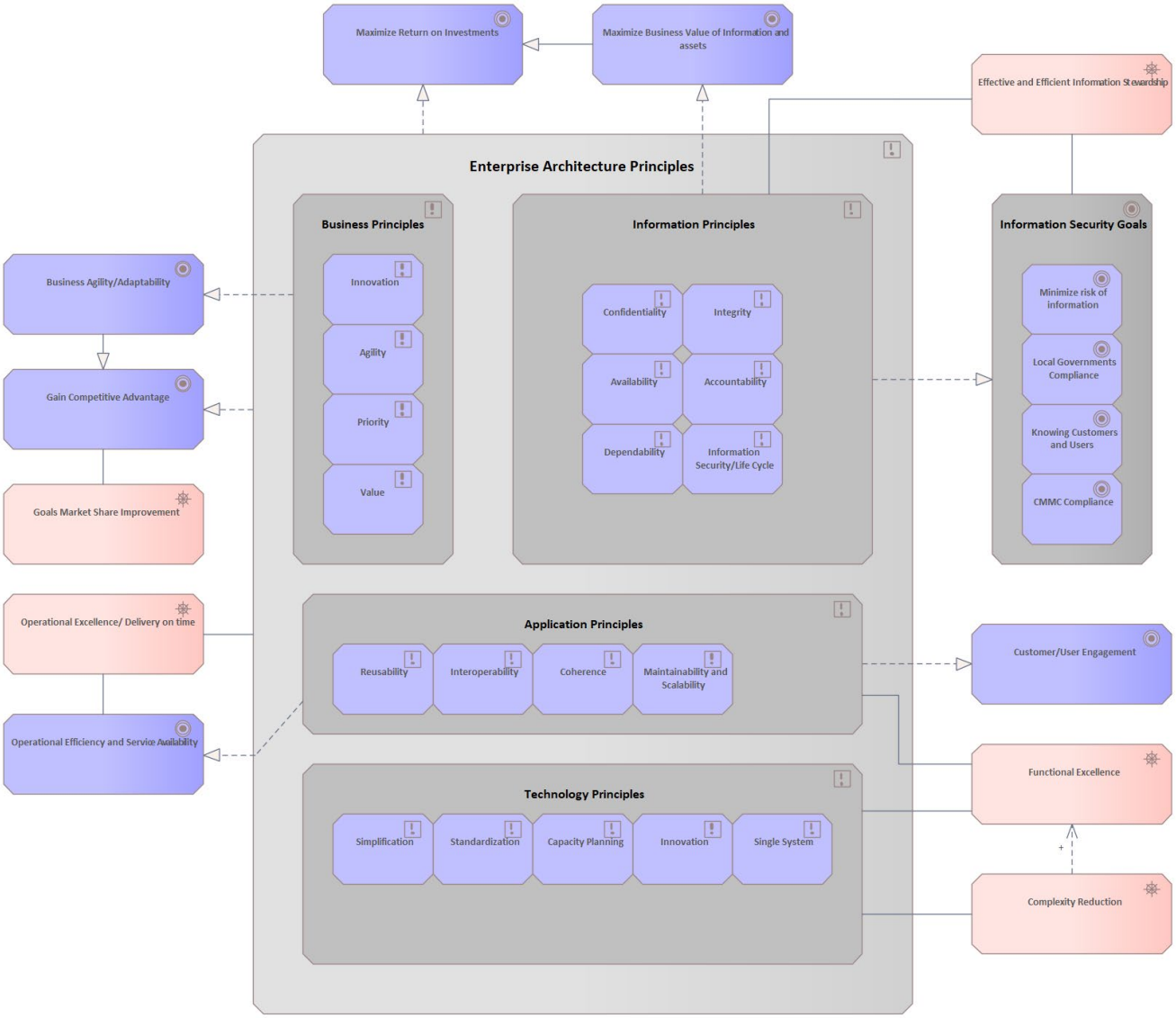
The practices, tools, principles, and techniques are built upon critical and invisible routines of thinking, behavior and cultural influences. There must be a synchronized adjustment between those two levels, to achieve and/or generate continuous improvement and adaptation to the unstoppable changes in the business environment, market and technology.

Figure 2: Practices, Tools, Principles and Management thinking (Rother, 2009)



Architecture principles should be setting boundaries, but the idea that the architecture role is mainly to set principles and guidelines for governing the design, must be discarded. There have been many cases in which beautiful and logical principles have given disastrous results.

Figure 3: Architecture Principles Sample



These architecture principles must do at least two things: prevent chaos and direct the enterprise into a much better situation. Setting boundaries could be unpopular, and this unpopularity could make it difficult for the business to truly value the contributions of the enterprise architects. Therefore EA should not be limited to a set of limitations but to the design, description and delivery of solutions. Not in all detail, but only those of relevancy.

"An architecture must be a tangible contribution to the result, instead of something of which the value is so far removed from the actual issues that nobody — except the enterprise architect — recognizes it as a constructive contribution. Rules in the physical architecture world, like building decrees or the utopian city-planning principles of Le Corbusier, do not lead to good designs by themselves". (Wierda, Chess and the art of Enterprise Architecture, 2015)

"A good chess player does not follow the rules, the rules are 'embedded' in his skill without being exactly visible during play. For enterprise architects the same must be true: the architect should not be governed by

the rules, the rules should be an (invisible) part of the architect's skills". (Wierda, Chess and the art of Enterprise Architecture, 2015)

"If principles and guidelines are generally not effective to produce enterprise architecture's results, should we abandon them? The answer is no, they remain useful in several ways. It is the enforcing of principles and guidelines as a design method instead of making collaborative design choices that is dangerous. Using principles and guidelines as prescriptive rules is what is risky. Using guidelines as descriptive rules for the organization can be useful, however. But we should drop the term '(design) principle' in most cases". (Wierda, Chess and the art of Enterprise Architecture, 2015)

To gain competitive advantage, businesses must not depend so much in the solutions themselves (whether they result from applying SAFe, ITIL, DevOps, TOGAF, ODM or any other framework/technique), but in the ability of the organization to understand conditions and create fitting, smart solutions. As (Rother, 2009) affirms, "the capability for continuous, incremental evolution and improvement represents perhaps the best assurance of durable competitive advantage and company survival."

Current and Future State Architectures

Because the future is an unknown factor, the efforts, should focus on achieving an effective and efficient system of management based on collaboration, for keeping the company adjustable and adaptable to unpredictable and dynamic conditions while its products and services surpass all customer expectations.

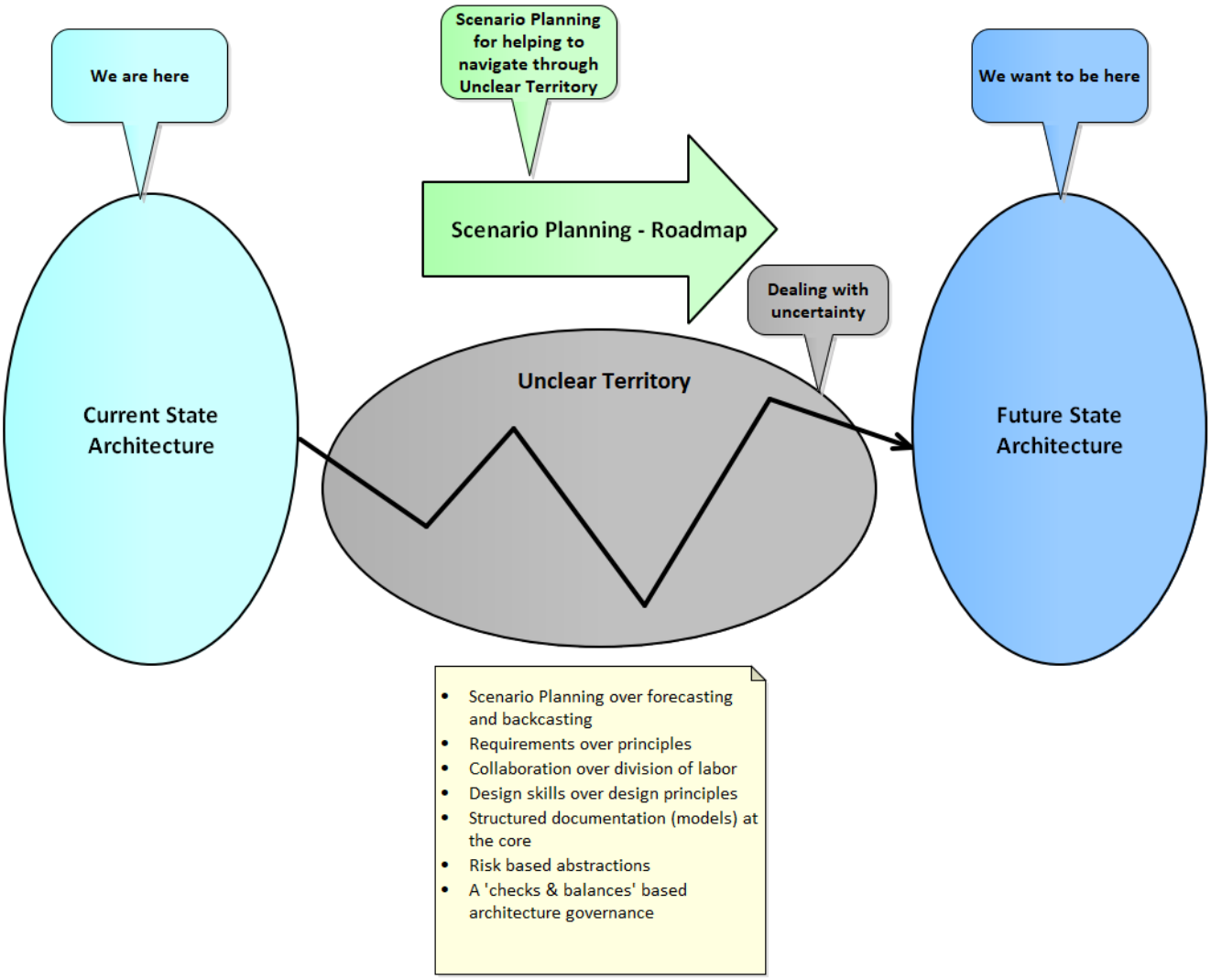
"Any organization's competitiveness, ability to adapt, and culture arise from the routines and habits by which the people in the organization conduct themselves every day. It is an issue of human behavior". (Rother, 2009)

"Perhaps there are three things that could be known with certainty: where we are, where we want to be, and by what means we should maneuver the unclear territory between here and there" (Rother, 2009). This affirmation must be taken in the dynamic context where companies move.

"If someone claims certainty about the steps that will be implemented to reach a desired destination, that should be a red flag to us. Uncertainty is normal—the path cannot be accurately predicted—and so how we deal with that is of paramount importance, and where we can derive our certainty and confidence". (Rother, 2009)

Something very important is to be convinced that there must not be "finish line" mentality. The following diagram is showing what is described in the previous paragraphs:

Figure 4: Current and Future States Architectures

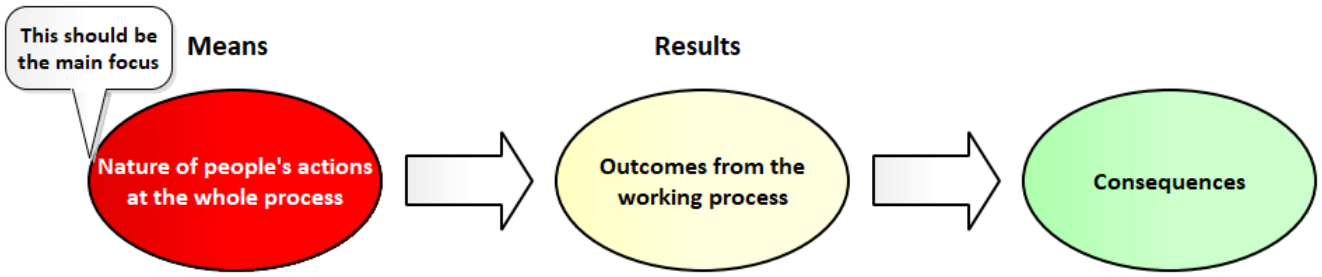


"The objective is not to win, but to develop the capability of the organization to keep improving, adapting, and satisfying dynamic customer requirements". (Rother, 2009)

"Any organization whose members can face unpredictable and uncertain situations (which are the norm) with confidence and effective action, because they have learned a behavioral routine for doing that, can enjoy a competitive advantage". (Rother, 2009)

Corporations must focus on the means and target conditions:

Figure 5: Means and Results



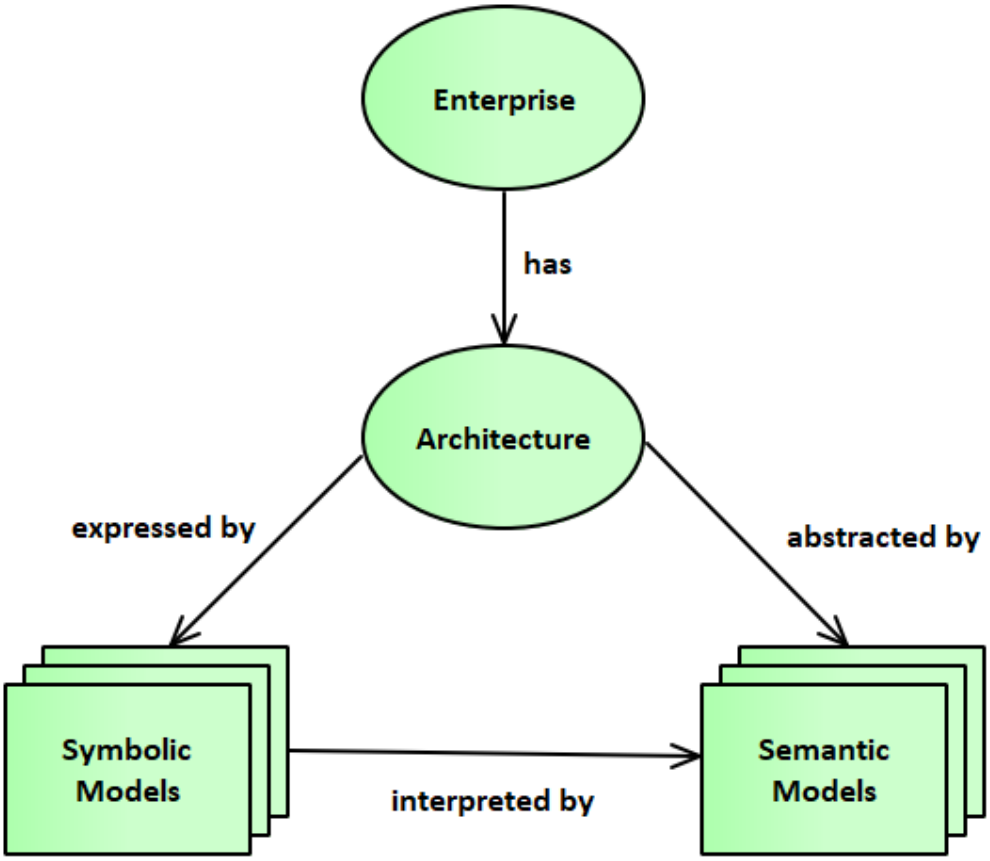
When developing IT projects, the leadership deals with the triangle time-budget-result, and in many cases, quality is reduced under pressure of time. This must be avoided. The means and target condition should guide the process. A target condition should be organized understanding the current position, through observation and analysis, coupling these situations with the direction, target, or need of the moment.

Modeling

This is a very important topic and vital for EA. I will dedicate a future paper to this.

"In many engineering disciplines, modelling a system consists of constructing a mathematical model that describes and explains it. In the fields of enterprise and software architecture, however, there is an overwhelming tendency to see pictures and diagrams as a form of model rather than as a form of language, or to be more precise, as a form of structure that helps in visualizing and communicating system descriptions. In other words, in architecture there is a tendency to replace mathematical modelling by ad hoc visualizations". (Lankhorst, 2016)

Figure 6: Symbolic and Semantic Models (Lankhorst, 2016)



Something to note is that the difference between formal and informal models is not the same as what is described above.

“Enterprise Architecture Modelling is a technique used to graphically and textually document the whole organization, not just its computer systems and processes, in a consistent, unambiguous, structured, maintainable manner”. (Brown, 2015)

Models can be created across varying degrees of detail based on the audience to whom the model is being presented. Models provide a common language for communicating enterprise architecture within an organization. Current state architecture, future state architecture, transitions/ gap analysis can all be represented through models across all the architecture domains.

EAs at any organization, should start using structured languages for modeling. There are many of them already well established and they need to be chosen according to the situation and/or needs.

(Wierda, Chess and the art of Enterprise Architecture, 2015) recommends modeling based on the ArchiMate structure, leading into a division of the following layers:

- Business and information-architecture (what does the organization do, and with what (abstract) information)
- Application- and data-architecture (how are the business processes supported by applications and how is business information represented as data in applications)

- Infrastructure architecture (how are the applications supported by infrastructure and how is application data represented as low level artifacts such as databases)

(Lankhorst, 2016) provides the following recommendations for modeling:

- A model must provide answers to questions.
- Make a clear distinction between a model and its visualizations.
- Maxim of Quantity:
  - Make your model as informative as necessary.
  - Do not make your model more informative than necessary.
- Maxim of Quality:
  - Do not model what you believe to be false.
  - Do not model that for which you lack adequate evidence.
- Maxim of Relevance:
  - Be relevant (i.e., model things related to the modelling goal).
- Maxim of Manner:
  - Avoid obscurity of expression.
  - Avoid ambiguity.
  - Be brief (avoid unnecessary concepts and relations).
  - Be orderly.
- Model iteratively
- Model for dynamics.
- Be economical in models.
- Be economical in views.
- Make concepts recognizable.
- Make structures recognizable.
- Make a model consistent.
- Keep related models consistent.
- Make models as correct and complete as needed.
- Treat different concerns orthogonally (independently).

Long Term Vision

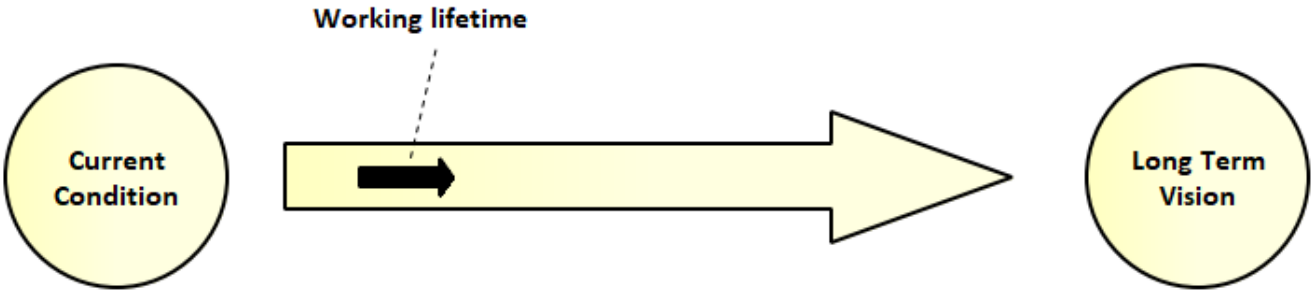
These long-term vision sample goals:

- Zero bugs/defects for Software Development
- No systems time down (IT Systems)
- 100% customer satisfaction
- The most innovative company in the business

May appear theoretical and not achievable, but it should not matter; those goals must serve to give direction, not to be discussed; and all efforts must be spent trying to move closer to them, continually improving, adapting, nurturing processes, and products into profitability.

Long term vision, or direction giver: “moving toward a desired state through an unclear territory by being sensitive to and responding to actual conditions on the ground”. (Rother, 2009).

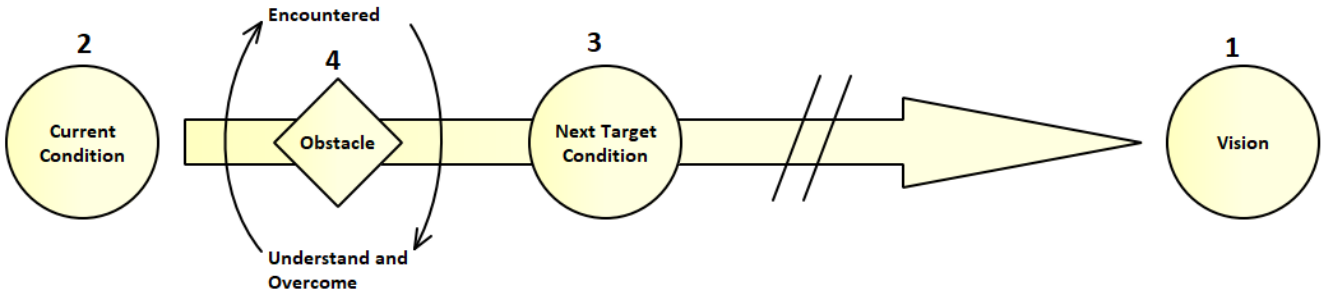
Figure 7: Long Term Direction Giver (Rother, 2009)



The sense of direction, as (Rother, 2009) explains, helps to use the cost/benefit analysis in a different way, not only to decide the financial feasibility of products/projects on a case by case basis, but rather to look at them in an integral manner, and to concentrate on how they may help the organization to get closer to the long term vision goals.

The cost/benefit analysis should be used mainly not to decide whether something should be done, but on how to do it. The long-term vision must drive the decision about what to do. The firm does not need to be perfect, just ahead of the competition in products and services.

Figure 8: Continuous Improvement (Rother, 2009)



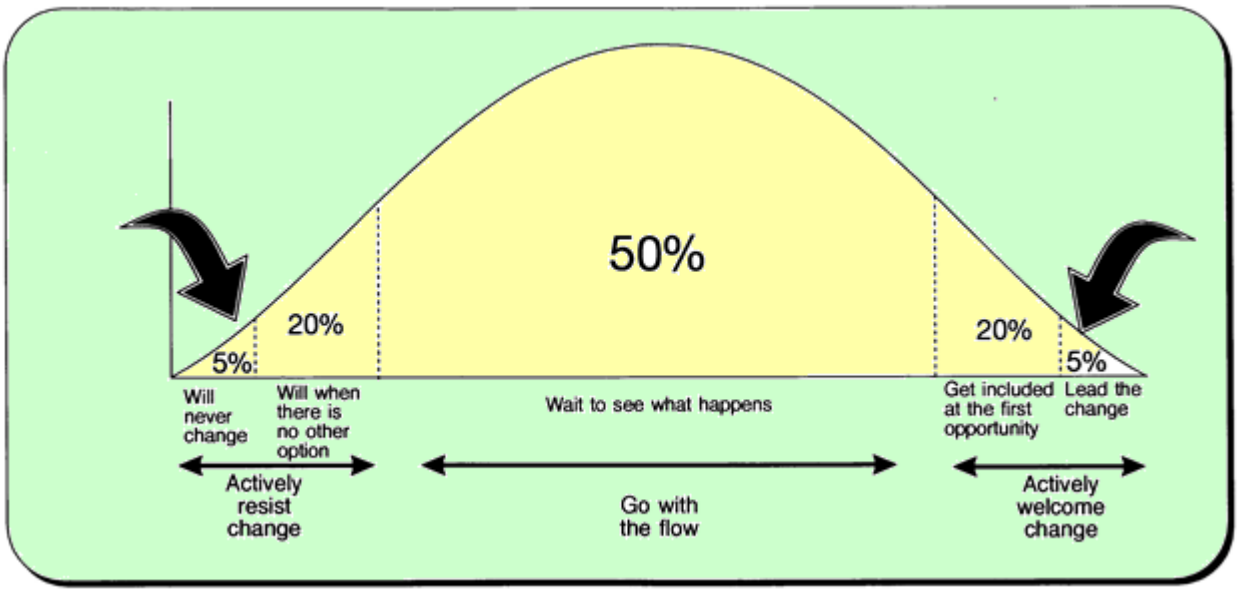
Change

Resistance to change, defined as actions taken by individuals and groups when a threatening perception of change is occurring in the environment where these persons operate. No Corporsation is immune to this kind of opposition.

“George Koenigsaecker, an early lean thinker in the United States, has depicted this effect using the normal distribution curve”. (Rother, 2009)

A version of that curve is following:

Figure 9: Reaction to Change



"What Mr. Koenigsaecker's diagram suggests is that only a small percentage of people in the organization (the right tail of the curve) will welcome a change effort and actively participate. Another small group (the left tail) will fight it actively. And the great majority—although they may nod and indicate their support—will be on the fence and waiting to see what is going to happen". (Rother, 2009)

Understanding that the changes, in this case, must be push down from the Senior Leadership; enterprise architects should be among the main agents and evangelists for this movement, setting up effective and efficient collaboration between all stakeholders.

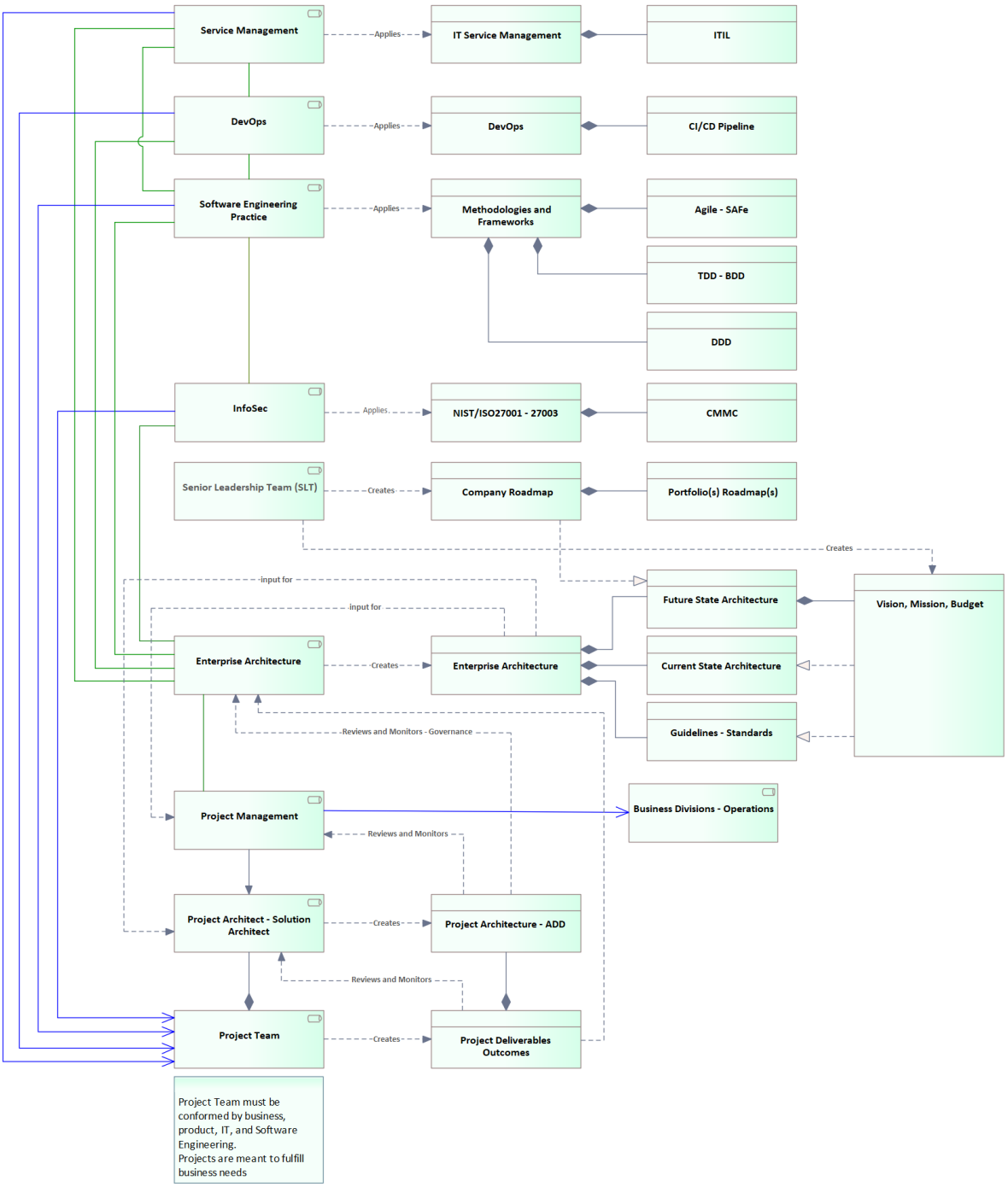
Long term mindset, collaboration, service management, DevOps, agile practice, EA, principles; should not be only the responsibility of groups, but something immersed at all levels of the organization directly proportional to their involvement in the decision-making process and their day-to-day activities.

Therefore, the EA role should contain the following capacities:

- analytical.
- complexity handling
- modeling
- learning (new businesses and technologies) fast
- independence and steadfast
- daring to go against the stream when necessary
- understanding the problems of others and acting responsibly towards them
- estimate the effect of scaling on a design decision such as model patterns
- finding relevancy
- finding root causes of IT problems quickly
- clearly explaining complicated issues
- charming personality.

These capabilities should be used as (Wierda, Chess and the art of Enterprise Architecture, 2015) expresses, to create an end result move by move, becoming realists, applying those skills for cooperation and bringing into existence an organization environment to handle complexity and unpredictability.

Figure 10: Roles and Objects



The diagram above illustrates the architectural context of a project focusing on the roles that should be part of it. These roles not necessarily have to be played for independent persons but would be better if the project team understands and applies them as if they were embedded in every member. The role of enterprise architecture in projects can be the same independently of a waterfall or agile approaches.

Working with the teams implementing changes is not smooth and easy, therefore the EA organization must have power to make sure it cannot be ignored, this power must be used to enable influence. EAs should not wield power but use the possibility of influence that comes with it. Trying to build influence without real power is risky, to say the least.

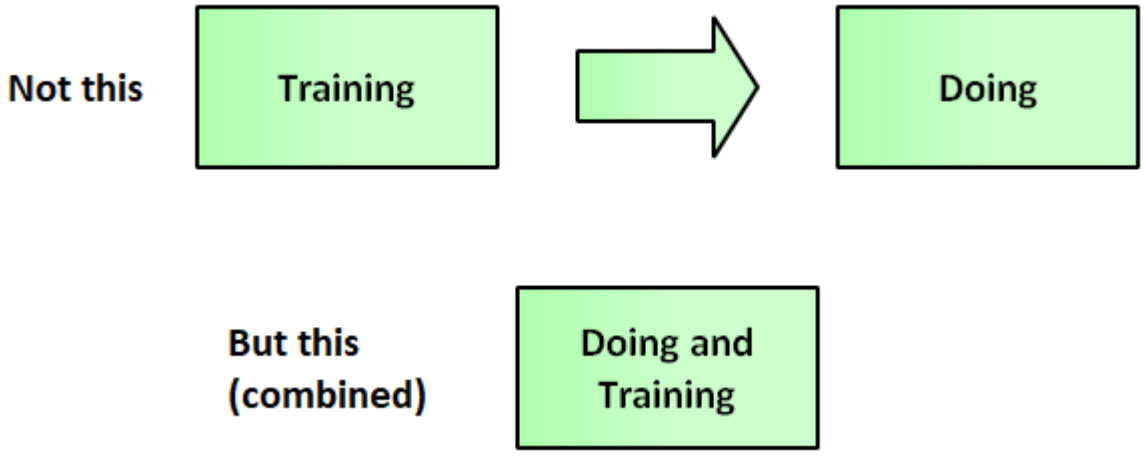
“Building influence is akin to making it more worthwhile for others to work with you than around/against you. One can see an organization that is to be influenced in terms of fluid mechanics. The organization, like a fluid, seeks the way of least resistance. So, getting influence requires two simultaneous types of activity: you need to make working with you as attractive as possible (pull), while making working around/against you as unattractive as possible (push)”. (Wierda, Chess and the art of Enterprise Architecture, 2015)

As mentioned before, to overcome complexity, collaboration is required. All stakeholders must see the Company’s long term goals as their own, and EA as an enabler. Isolationism that degenerates into a ‘siege mentality’ kills collaboration.

How do we change?

“The field of psychology is clear on this: we learn habits, automatic reactions, by repeatedly practicing behaviors. To build new mental circuits, we must practice a desired behavior pattern and periodically derive a sense of achievement from that behavior”. (Rother, 2009)

Figure 11: How to change?



“Intellectual knowledge alone generally does not lead to change in behavior, habits, or culture. Ask any smoker”. (Rother, 2009)

“It is now clear our problem was the level of analysis. As Albert Einstein famously remarked, “The significant problems we face cannot be solved by the same level of thinking that created them.” The problem with our efforts to understand IT architecture was that the level of analysis was all wrong. The focus needs to be higher—on enterprise architecture, the organizing logic for core business processes and IT infrastructure reflecting the standardization and integration of a company’s operating model. We have come to understand that enterprise architecture boils down to these two concepts: business process integration and business process standardization. In short, enterprise architecture is not an IT issue—it’s a business issue”. (Ross, Weill, & Robertson, 2006)

“The breakthrough in our understanding lay in an apparent contradiction. In a business world that is changing faster than ever before, the top-performing firms create a stable base—they digitize their core processes and

embed those processes into a foundation for execution. This stable foundation makes a company both more efficient and more agile than its competitors". (Ross, Weill, & Robertson, 2006)

Future

The future is uncertain, but somehow Organizations need to create a people and technology ecosystem that will be adaptable and capable to handle unpredictability while gaining competitive advantage.

Artificial Intelligence, Machine Learning, Robotic Process Automation, Blockchain, Virtual Reality, Internet of Things, Cyber Security, Data Analytics, Microsystems, Cloud Technology, etc., are among the new technologies coming in the last years and many others will come.

Quantum computing is one of them, and it promises a huge revolution. Shall Enterprises need to look at it now?

(Microsoft, 2019), (IBM, 2019), and others are already working and looking at the possibilities and applicability of this new way of computation.

"All computations involve inputting data, manipulating it according to certain rules, and then outputting the final answer. For classical computations, the bit is the basic unit of data. For quantum computations, this unit is the quantum bit—usually shortened to qubit". (Bernhardt, 2019)

"Quantum computation is not a new type of computation but is the discovery of the true nature of computation". (Bernhardt, 2019)

Action Plan

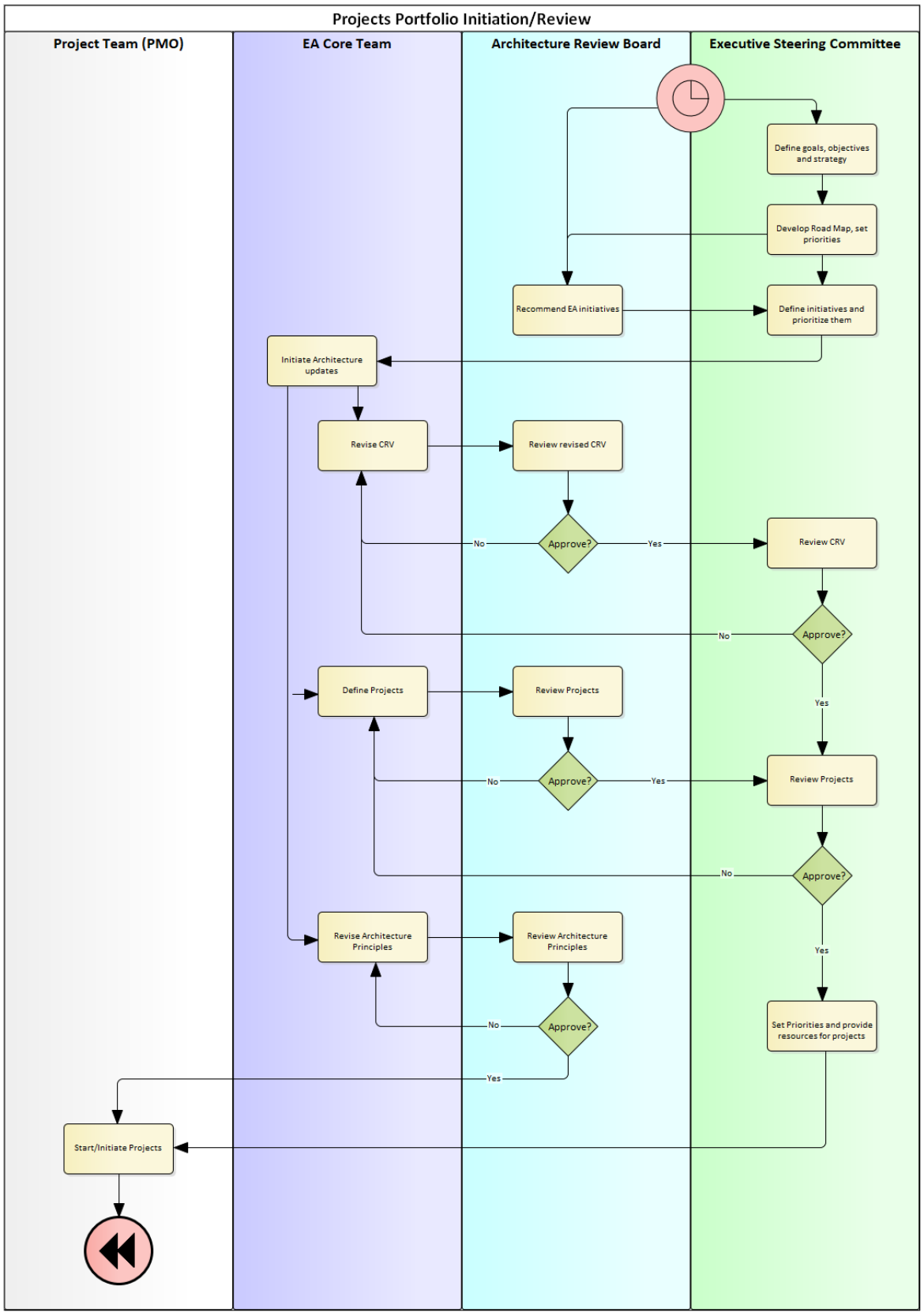
Businesses need to pursue an organization that facilitates continuous improvement and adaptiveness to successfully compete and win. It is necessary to achieve business process integration and standardization through collaboration. This is illustrated by Figure 9: Long Term Direction Giver (Rother, 2009)and Figure 10: Continuous Improvement (Rother, 2009)

The corporate culture needs to change from isolationist and political into something that is required to be successful, collaboration. The most damaging of all the ways collaboration can be prevented is the extreme form of isolationism that (Wierda, Chess and the art of Enterprise Architecture, 2015) calls 'siege mentality'. Therefore, any case of this attitude must be eradicated.

The only way enterprise architecture can be effective in addressing real problems is by getting all support from the top management, and setting up effective coordination between all stakeholders, by departments, by projects, etc. Basically, all stakeholders must see enterprise architecture as part of their own interests. This is not automatically the case, but EA must work to achieve it.

The EA practice, should start adapting, refining and applying the project architectural context presented in Figure 12: Roles and Objects. Initiatives and projects might follow the ideas from these suggested processes:

Figure 12: Project Portfolio Initiation/Review Process



CRV: Common Requirements Vision

Excellence cannot be achieved without good training and tools. Companies need to invest in both. Training needs to be at all levels, keeping in mind what is exemplified with Figure 13: How to change?. Tools for planning, designing, and testing are to be acquired and/or updated.

As important as the Agile Manifesto, the following principles (Wierda, Chess and the art of Enterprise Architecture, 2015), should guide The EA practice (Figure 5: Current and Future States Architectures):

- Scenario planning over fore and back casting
- Requirements over principles
- Collaboration over division of labor
- Design skills over design principles
- Structured documentation (models) at the core
- Risk based abstractions
- A 'checks & balances'-based architecture governance

The risky environment of the businesses make it necessary to focus on the scenario planning for the construction of the road maps and future state architectures. "If we look back over the history of planning in organizations, we can see a fundamental illusion that is beginning to come to light. The illusion is that planning can function like a machine, that the steps of organizational planning need only be carried out. Planning therefore needs to take a different approach, one that assumes tomorrow's world will be fundamentally different from today's one. Scenario planning explores a variety of outcomes, a variety of potential answers, and uses them to create awareness and readiness". (Chermack, 2011)

"Causal Texture Theory (CTT), proposed by Emery and Trist and based on the field of socio-ecology, sets out to map the various changes as they observed these to include the human element. This required mapping out the various ways in which people saw strategic planning in relation to the way that the new turbulent contextual environment is understood. They decided to use connectivity as an organizing principle. Specifically, connectivity between the internal organizational world over which the strategist has some control, and the external world, which constitutes the environment that is outside any control of the strategist". (Ramirez & Wilkinson, 2016)

It is common knowledge that external environments affecting all organizations, have different levels of the following characteristics:

- Volatility
- Uncertainty
- Complexity
- Ambiguity

The industries face multiple risks, among them:

- Terrorism
- Sharing economy
- Disease, like COVID, Ebola, Yellow Fever, etc. have had detrimental impact

Consequently, scenario planning using artificial intelligence and other available technologies and theories, should be applied. The next figures show what would need to be adapted.

Figure 13: Theory of Scenario Planning (Chermack, 2011)

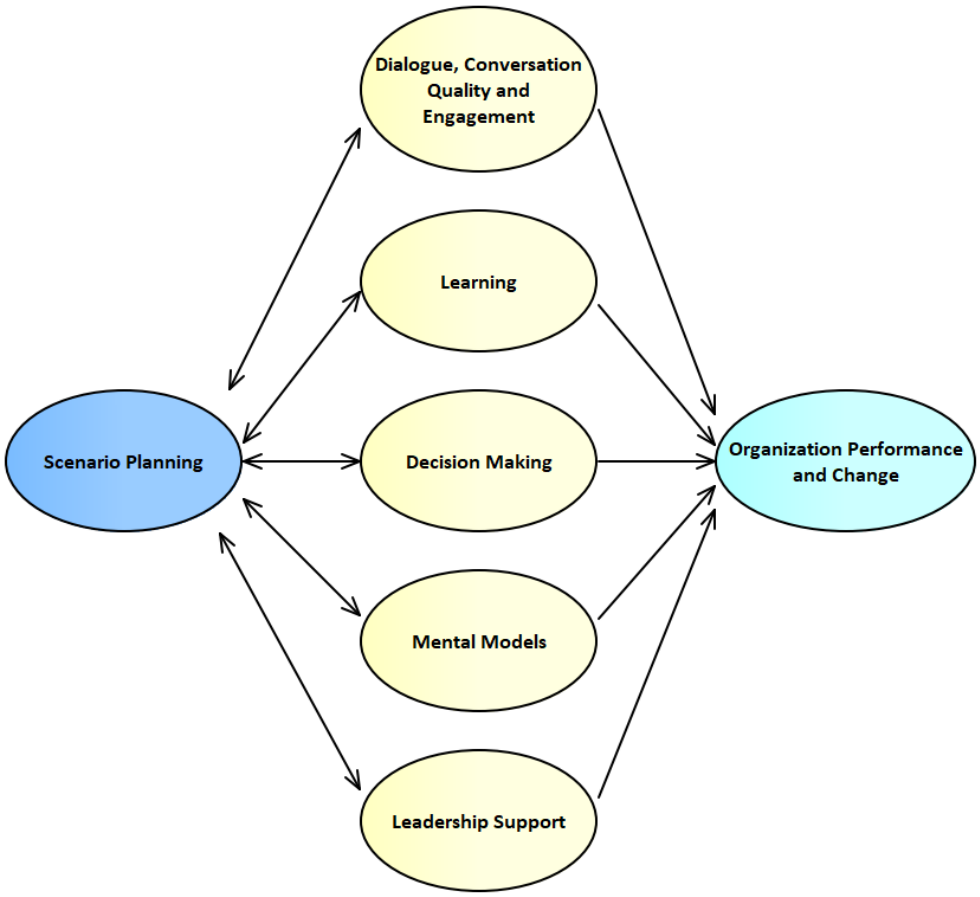
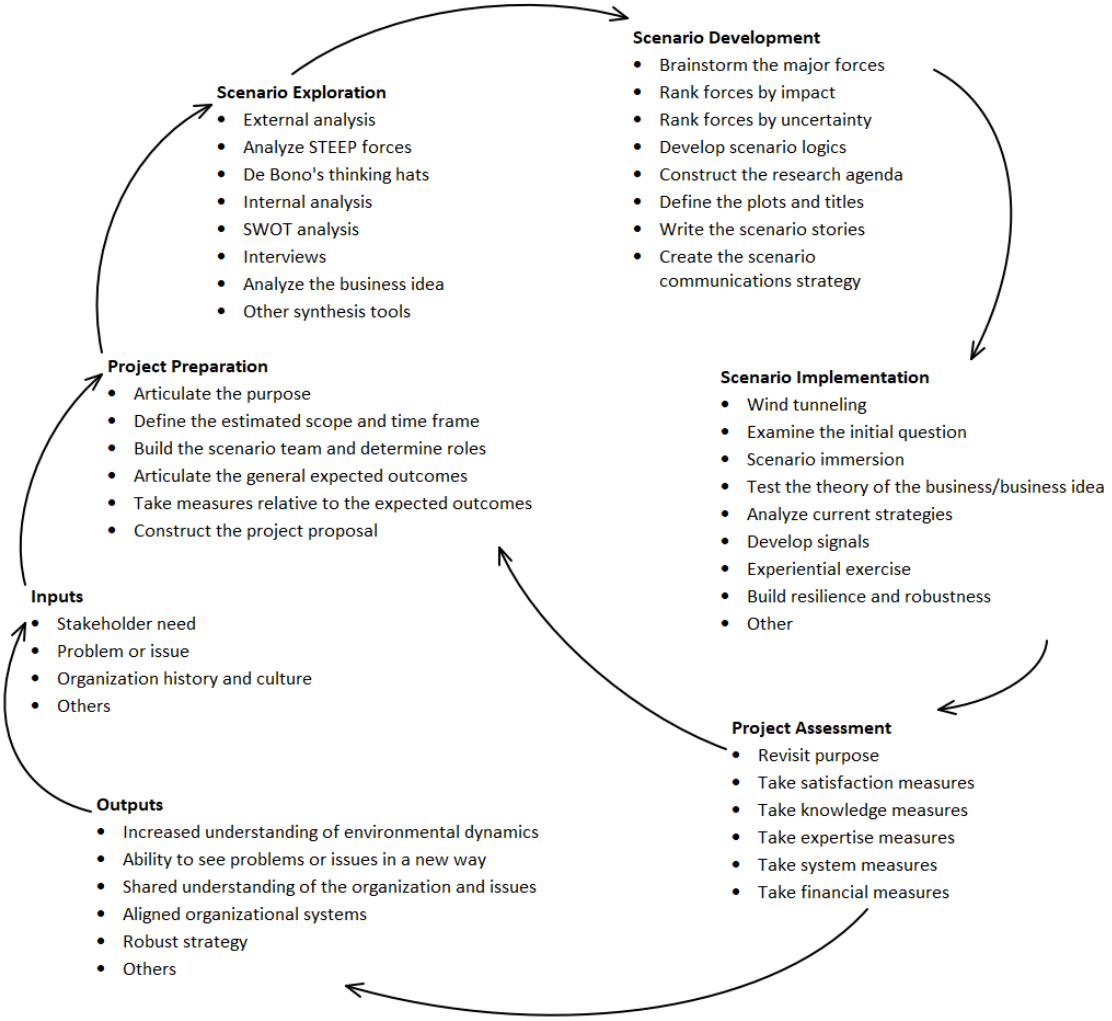


Figure 14: Performance Based Scenario (Chermack, 2011)



References

Bernhardt, C. (2019). *Quantum Computing for Everyone.* The MIT Press.

Brooks Jr., F. P. (2010). *The Mythical Man-Month, Anniversary Edition.* Addison-Wesley, Pearson Education.

Brown, M. T. (2015). *Understand your Organization - Improve your Business* . Bite-Sized Books Ltd.

Chermack, T. J. (2011). *Scenario Planning in Organizations: How to Create, Use, and Assess Scenarios.* Berrett-Koehler.

Colorado State University. (n.d.). *Scenario Planning Institute*. Retrieved from Colorado State University: https://www.chhs.colostate.edu/spi/

Desfray, P., & Raymond, G. (2014). *Modeling Enterprise Architecture with TOGAF.* Morgan Kaufmann.

Emmeche. (1997). Aspects of Complexity in Life and Science. *Philosophica vol. 59*, 41 - 68.

Group, T. O. (2018). *The Open Group Standard, Version 9.2.* The Open Group.

IBM. (2019). *Theory of quantum computing and information group*. Retrieved from IBM Research: https://researcher.watson.ibm.com/researcher/view_group.php?id=5881

Kim, G., Humble, J., Debois, P., & Willis, J. (2016). *The DevOps Handbook.* Portland, OR: IT Revolution.

Lankhorst, M. e. (2016). *Enterprise Architecture at Work - Fourth Edition.* Springer.

Leffingwell, D., Knaster, R., Oren, I., & Jemilo, D. (2018). *SAFe 4.5 Reference Guide* . Scaled Agile, Inc.

Microsoft. (2019). *The Q# Programming Language*. Retrieved from Microsoft Docs: https://docs.microsoft.com/en-us/quantum/language/?view=qsharp-preview

Pink, D. (2010, April i). *RSA Animate*. Retrieved from YouTube: https://www.youtube.com/watch?v=u6XAPnuFjJc

Ramirez, R., & Wilkinson, A. (2016). *Strategic Reframing.* Oxford University Press.

Rao, P. C., Reedy, A., & Bellman, B. (2019). *Certified Enterprise Architect.* Mc Graw Hill Education.

Ross, J. W., Weill, P., & Robertson, D. C. (2006). *Enterprise Architecture as Strategy.* Harvard Business School Press.

Rother, M. (2009). *Toyota Kata Managing People for Improvement, Adaptiveness, and Superior Results.* McGraw Hill Education.

Scaled Agile, I. (2019). *SAFe*. Retrieved from Scaled Agile Framework SAFe: https://www.scaledagileframework.com/

Townson, S. (2008, November). Why does Enterprise Architecture Matter?

Wierda, G. (2015). *Chess and the art of Enterprise Architecture.* R&A.

Wierda, G. (2017). *Mastering Archimate.* R&A.

Wikipedia. (2019). *ISO/IEC 42010.* Retrieved from Wikipedia: https://en.wikipedia.org/wiki/ISO/IEC_42010

Wikipedia. (n.d.). *NIST Enterprise Architecture Model*. Retrieved from Wikipedia: https://en.m.wikipedia.org/wiki/NIST_Enterprise_Architecture_Model

Yakyma, A. (2016). *The Rollout.*