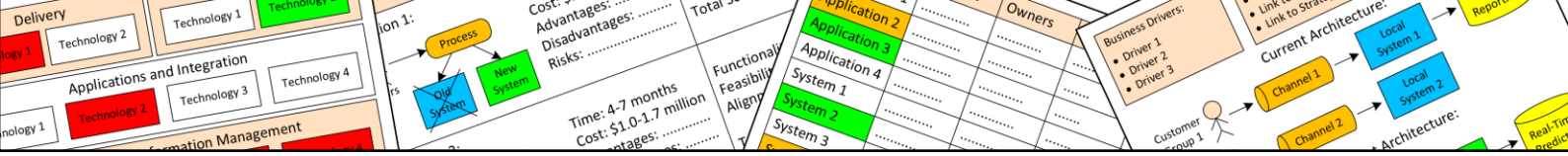


Interview with Svyatoslav Kotusev by Shen Liu (Thoughtworks China)

Svyatoslav Kotusev (kotusev@kotusev.com)

Published in November 2022 by Thoughtworks China
URL: https://insights.thoughtworks.cn/interview-svyatoslav-kotusev/ (in Chinese)

Introduction

Q1: Could you please introduce yourself to Thoughtworks' friends?

In short, I am a researcher in the discipline of enterprise architecture (EA). I have been doing my studies on a full-time basis since 2013. Among the products of my research are the best-selling book *The Practice of Enterprise Architecture*, around 50 academic and practitioner articles on EA, as well as some other helpful materials for enterprise architects, like *Enterprise Architecture on a Page*. More information about my research and its outputs can be found on my website http://kotusev.com.

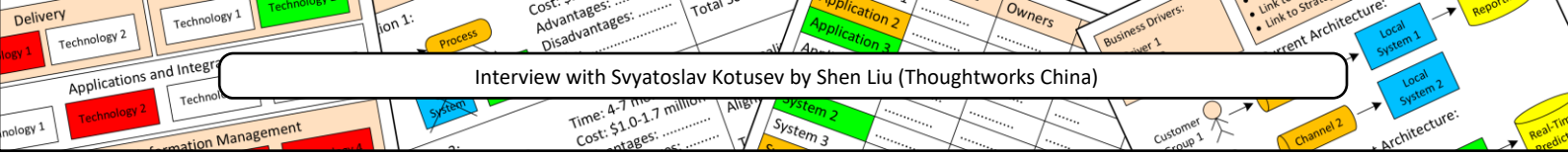
Why Learn Enterprise Architecture?

Q2: At what moment did you get in touch with EA and decide to start researching EA?

I began my career in the industry as a software developer, then progressed to a software architect and wanted to evolve further to become an enterprise architect. However, along the way, I realized that researching and analyzing things is much more fascinating to me than actually doing them. Therefore, I have swerved from the industry to academia and became an EA researcher instead of a practicing enterprise architect.

Q3: What exactly is an EA (although your book mentions it)?

Indeed, the very term "enterprise architecture" has countless definitions and numerous meanings have been ascribed to it, making it tremendously difficult to say exactly what EA is, landscape structure, planning approach, decision-making philosophy or merely a bunch of architectural documents. In my own writings, I tend to define EA in a traditional way as a collection of documents facilitating joint business and IT planning. However, I also use a number of complementary terms for associated notions: "EA practice" for the organizational practice of planning, "EA artifacts" for concrete architectural documents, "EA function" for the organizational unit responsible for planning and "EA discipline" for the whole body of knowledge on EA. Importantly, my system of definitions is by no means the only possible or the best one, but it is internally consistent and allows clearly referring to specific concepts related to EA, rather than to some vague abstraction called "enterprise architecture".



Q4: The Open Group published an article a while ago "Enterprise Architecture Is a 'Foundation Skill' for the Engineering Students". Thoughtworks also recruits a large number of engineering students every year. What do you think is the use of EA for students? If it does not help much, why?

In my personal opinion and observations (but I do not pretend to be authoritative on this matter), university students without practical experience in the industry should be provided with only a cursory introduction to the concept of EA. This is because most undergraduate students, at least those that I have met, barely understand the complexities of large organizations and their IT landscapes and, thus, can hardly appreciate the role and value of EA. Besides that, understanding EA is certainly not about studying TOGAF, IT4IT and ArchiMate standards, as The Open Group article suggests.

Q5: Thoughtworks is an IT consulting company, we mainly do IT consulting and delivery projects. Each project will include various roles, such as Business Analyst, Program Manager, Product Manager, QA, UX, Tech Lead, Developer, Data Scientist, and more. Which type of role do you think should learn and master EA the most and why?

Interestingly, nowhere in your list of roles did you mention any architects (e.g. solution architects directly involved in the project work) who should be the primary candidates for learning EA. Because none of the roles you listed is architectural, I believe the best answer to your question could be this: people like program managers, product managers, tech leads and developers should gain awareness of EA to understand that every project in an organization is implemented within the broader organizational context and, thus, should take into account certain enterprise-wide considerations, rather than being driven only by local motives.

For example, taking shortcuts to meet the specified deadlines may please project managers and product owners, but eventually turn out harmful to the company due to architecturally suboptimal, shortsighted choices that will have to be redeemed in the future. Or, allowing developers to select their own favorite technologies for project implementation may be a good approach in the short term, but it will certainly lead to major problems for the company in the long run related to uncontrolled technology proliferation.

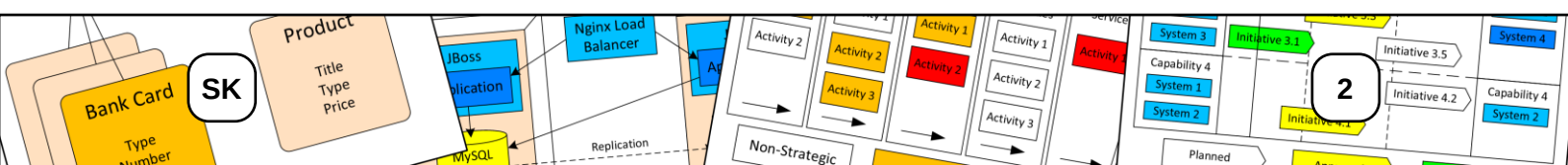
In short, various non-architectural project participants should learn EA, for the most part, to realize the existence of the overall organizational context and respect global concerns dictated by this context in their work and decision-making. That would be beneficial to organizations.

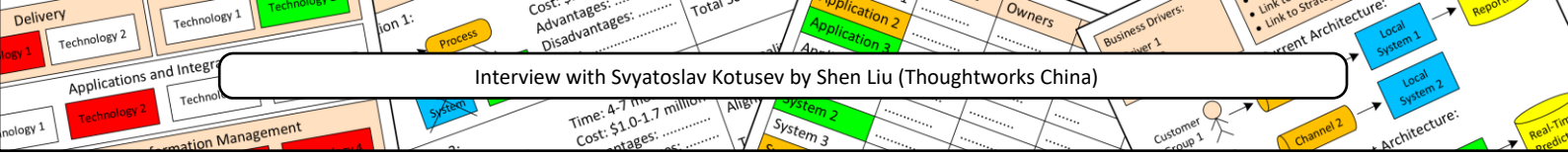
How to Learn Enterprise Architecture?

Q6: What do you think is a better path to learn EA from scratch?

For a basic understanding of what EA is all about, which can be sufficient for project managers, software developers and the like, I believe it would be enough to read my book or study other available materials on the subject. However, if one wants to become an enterprise architect, then I am afraid that any formal education will be insufficient because working as an architect requires well-developed *behavioral* skills, most importantly those that can be best summarized in the word "communication". These skills can be acquired only through practice.

For this reason, in my view, the only real way to become an architect is to observe what experienced architects do, how they do it and keep an eye on their "tricks of the trade". In short, to become an architect, learn from other architects. One important implication of this





circumstance is that one simply cannot learn the craft of EA while working in an organization where EA is not practiced, but this is a somewhat separate and rather long story.

Q7: For those with an engineering background and those without an engineering background, what do you think are the challenges for them in the process of learning EA? How to meet these challenges?

For people with an engineering background, the key challenge is arguably to realize that finding organizationally acceptable solutions is not at all an engineering task as it cannot be approached by rigorous analytical means familiar to them. Instead, reaching acceptable solutions in a heterogeneous stakeholder environment requires *political* approaches so alien to engineers. To cope with this challenge, engineers have to understand that the best solutions in the organizational context are *not* those solutions that seem most rational, but those that other people can agree with, and alter their attitude accordingly.

For people without an engineering background, the most significant challenge is arguably to realize that technology is complex, rigid and implies numerous intricate, hidden interdependencies that complicate virtually every attempt to change it. To overcome this challenge, non-techies should probably try to study any serious technology in detail to see with their own eyes how astonishingly complex the world of IT is and start paying due respect to technology-related concerns.

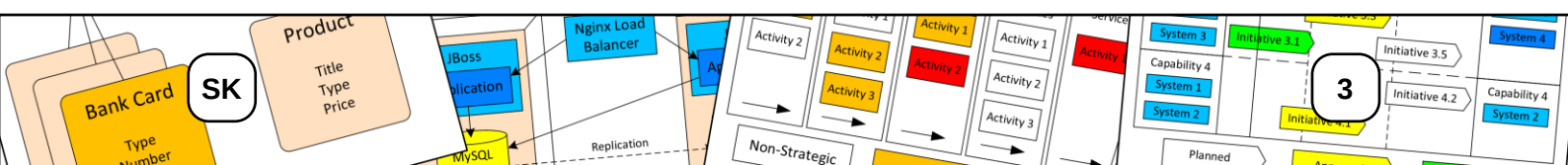
Interestingly, successful enterprise architects are both engineers and politicians at the same time, capable of designing decent technical solutions and also "selling" these solutions to the organization.

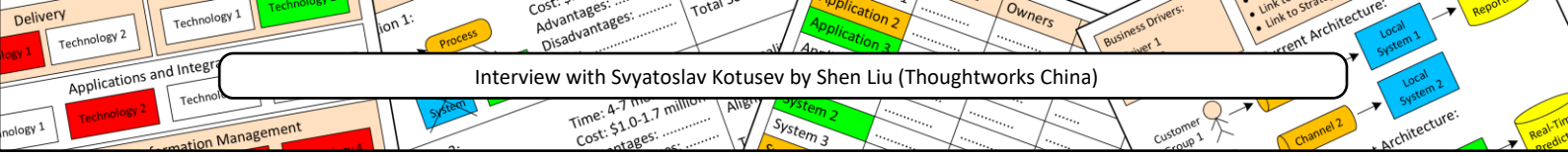
Q8: In addition to your website/book, can you recommend some good resources (website/video/book/journal) for learning EA, especially where we can find some real EA cases?

For a number of reasons, most available materials on EA are, in my view, surprisingly shallow and unworthy, serious analyses of real-world EA practices remain very scarce. Some of the sources that I consider informative include (please forgive me if I forgot to mention someone) the famous book *Enterprise Architecture as Strategy: Creating a Foundation for Business Execution*, the hidden-gem book *Managed Evolution: A Strategy for Very Large Information Systems* and other books like *Strategic Enterprise Architecture Management: Challenges, Best Practices, and Future Developments* and *Chess and the Art of Enterprise Architecture*. All these books analyze experiences of real organizations, not fictional ArchiSurance, and provide some valuable insights on different aspects of EA.

Q9: How to test the effectiveness of my learning of EA? Or, from another angle, how to test a person's EA ability and level?

This is a tremendously difficult question as neither the effectiveness of a single architect nor the effectiveness of the entire EA practice in an organization can be easily assessed, let alone measured. These things, arguably, can only be evaluated subjectively, based on a personal perception of individual or organizational effectiveness, particularly from the standpoint of achieved results. It is difficult to imagine any formal tests to estimate one's EA abilities, but one's observed successes and failures can serve as their indicators.





How to Apply Enterprise Architecture?

Q10: What is your opinion on the current EA framework? Could you use TOGAF as an example and talk about what you think are its strengths, limitations and problems?

In general, my opinion on the current EA frameworks (e.g. Zachman, FEAF, DoDAD and TOGAF) is best summarized in the title of one of my articles *Enterprise Architecture Frameworks: The Fad of the Century*. These frameworks rose to prominence on the wave of hype around EA, wasted time and money of organizations trying to use them and then became purely symbolic.

The situation specifically with TOGAF can arguably be best illustrated by the analogy to horoscopes. From a scientific perspective, horoscopes possess no predictive power whatsoever and, on this basis, should undoubtedly be considered useless. And yet, many people use horoscopes either because they prefer to believe in their predictions, or simply because they find horoscopes amusing, appeasing or comforting due to the false sense of certainty that they instill.

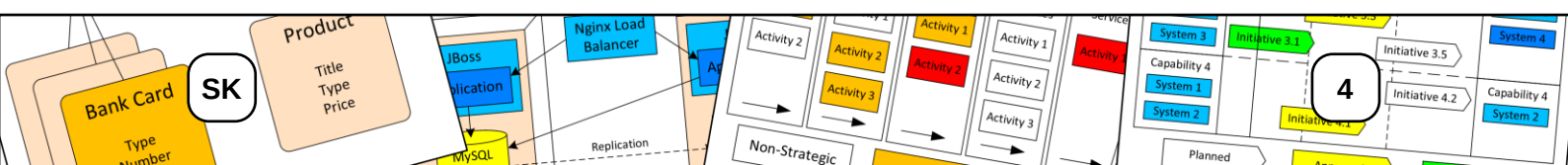
Exactly the same situation is currently observed with TOGAF. TOGAF has demonstrably no correlation with what happens in successful EA practices (except that some EA artifacts are indeed produced and other trifles) and, on this basis, should be considered useless. And yet, many people study it and try to interpret it "properly" to figure out what to do. The fact that real EA best practices are simply not there, surprisingly, does not stop the public from searching for the key to success with EA in TOGAF volumes. Like with horoscopes, TOGAF's evident and fundamental disconnection from reality does not forbid a person to believe in its utility, contrary to any logic and common sense.

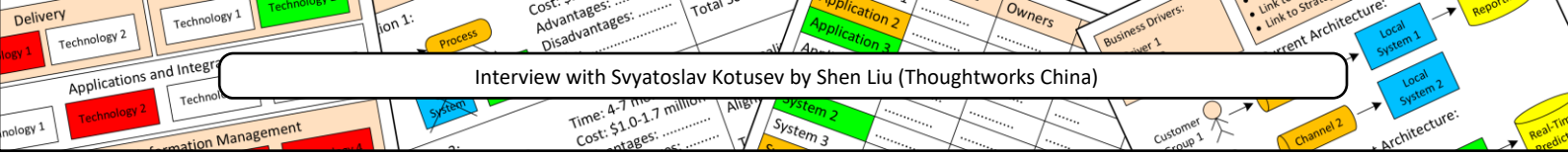
In this light, it is utterly meaningless to talk about TOGAF's "strengths" or "limitations", but only to acknowledge its single critical problem: the absence of any serious resemblance to empirical reality. The rest is mere speculations or coincidences.

Q11: Could you tell us the story of the CSVLOD model? How was this model formed? What are its advantages and disadvantages?

The story behind the CSVLOD model (the one that classifies all EA artifacts based on their roles in EA practices into six general types: Considerations, Standards, Visions, Landscapes, Outlines and Designs) is rather simple. When I started to study EA, I realized that existing models of EA (e.g. BDAT — business, data, applications and technology), first, cannot really classify EA artifacts as many of them cover multiple domains and, more importantly, do not explain their functions in the organizational context. My further analysis of EA artifacts and their usage in organizations has led to a conclusion that, despite their multiplicity and diversity, they all can be reduced to only six general types with a number of unique attributes, which I titled Considerations, Standards, Visions, Landscapes, Outlines and Designs (CSVLOD).

Regarding the advantages and disadvantages of the CSVLOD model, I would formulate them as follows. One advantage of the CSVLOD model is that, to my best knowledge, it is the only model of its kind that has originated from the analysis of factual empirical realities in organizations, rather than from "holy scriptures". Another advantage of this model is that it actually explains a lot about EA artifacts in terms of their informational contents, presentation formats, target audiences, usage scenarios, temporal lifecycles and many other properties. Of its disadvantages, like any model, it presents only an approximation of reality, or a simplification of it, and, in this sense, is inherently inaccurate.





Q12: After mastering the theory of EA, how to apply EA is a pain point in our projects. For example, we help a client deliver an "after-sales operation system". If the client has relatively clear needs, then according to our mature Agile project requirements and development process it can be completed. Is EA only useful for enterprise-level IT transformation projects?

By definition (albeit it has countless definitions), EA is mostly about the architectural work that happens at a level higher than separate projects, or before any projects are launched. So, if your task is to deliver a system with stipulated requirements (i.e. the project has already been shaped), then the approaches associated with EA are unlikely to be relevant. It means that the client has already applied some or the other EA-related techniques to figure out that it is this project that needs to be implemented and it is this vendor that should be involved to deliver it. Put it simply, in your example, the EA-type of work takes place at the client's side, not at the consultant's side.

However, the awareness of EA can help the consultant to please the customer by requesting its architectural principles, guidelines, patterns and landscape diagrams to align the new system with the client's standards and properly integrate this system into its IT landscape. Although the consultant does not use EA directly, the awareness of it can help the consultant do a better job for the customer. SK