

Enabling Agile Enterprise Risk Management Part II: Disciplines

Description

In [the previous article](#) (Enabling Agile Enterprise Risk Management: Artifacts and Disciplines,) I identified the artifacts necessary to enable you to *transform* your enterprise at speed. In this article, I will discuss the disciplines required to *transform* and not just *change*, which can create Technical Debt.

Artifacts That Document WHAT

The artifacts include your Enterprise and Business Architecture (EA and BA) models, your Business Process Management (BPM) models and your Risk Register. Two important elements not covered in the previous article are an Ontology and a Pattern Library, which integrate with the EA model and with each other. They ensure consistent naming and usage of object types and instances of the objects that appear in the models. This is intrinsic to Knowledge Management (KM) and asset and component reuse, and is fundamental to avoiding Technical Debt. For instance, if a business unit were in need of particular capability, an ontological classification tag in the EA model should ensure that decision-makers can consistently identify whether one already exists, identify where it's used and research whether it can be reused. If it exists, the Pattern Library should contain content or assets that point the way to accessing it, design documents for it or even copies of assets that enable it. Simply speaking, if you don't know what you have, where it is and what it's good for, you can't reuse it.

The first step in applying the EA, BA and BPM disciplines is to create and populate the current state models that embody them. The second is to implement work processes around them to keep them current as the enterprise evolves. The final step is to embed them and the information they contain into the design and implementation processes associated with transformation. The aspect of transformation to which these apply is **WHAT**. **WHAT** is what you are going to build, what your post-transformation organization will look like, which processes you will revise and what they will look like after you do.

Disciplines That Determine HOW

The disciplines that enable you to execute the process of *transforming* are Scenario Analysis and Roadmapping, Project Portfolio Management (PPM) and Program and Project Management (PgM/PM).

Scenario Analysis is a practice in which alternative future states are evaluated and the likelihood they'll occur applied to determine the future-state architecture that will best position the company to compete, given what is determined to be most likely. Roadmapping involves defining the route that will be pursued to achieve the architecture. Invariably, there will be more initiatives vying for approval than can be executed at any one time and Project Portfolio Management is applied to determine relative value, prioritize and ultimately select, approve, fund, staff and schedule those chosen for execution. Finally Program and Project Management disciplines are applied to execute initiatives predictably and efficiently. In essence, these disciplines are about **HOW**. **HOW** is the path you will take to arrive at the future state that you have chosen from among the alternatives you have considered. It encompasses all aspects of prioritizing, selecting, scheduling and executing your transformation initiatives.

Where Agile Enterprise Risk Management Fits In

I will not impugn the job risk managers do by questioning their competence but there is increasing peril in accumulating small issues that go unnoticed, especially when the rate at which things change is high. Clearly, the risk in a major overhaul of a company's order entry system would be pretty front and center but, perhaps, the potential impact of system updates on the platform on which it runs might not be, especially if the updates were most relevant to another application that shared the infrastructure with it. An opportunity to enter a new line of business by product extension could depend on an automated or semi-automated business process that supports another product line and could necessitate a major logic overhaul that could impact the cost-benefit outcome of the project.

Understanding dependencies vertically in the EA model hierarchy and identifying shared elements horizontally across Value Chains or shared capabilities is critical to uncovering things that might otherwise fall through the cracks. As is the case with all implementation or transformation projects, finding dependencies late in the game is high-risk and usually results in excessive and avoidable negative impacts to execution.

From an artifact and discipline standpoint, up-to-date models are a must. Out-of-date information is almost worse than no information as it may well lead to false assumptions, inappropriate designs and misaligned plans. Secondly, it is critical to understand where risks actually attach to the objects in the models. For instance, many risks attach to capabilities' enablers, not the capabilities, themselves, even though their impact might be perceived there. So, the need to upgrade a capability may result in increased risk in an enabler and this risk can propagate to other capabilities that share it.

None of this is entirely new. People have been changing and transforming businesses since there have been businesses and, most of the time, they are able to recognize and deal with the issues for which they must account. What is new is the level of compartmentalization and componentization inherent in today's digital businesses and the dispersion of computing resources, subscribed services and polyglot solutions that power them. What is also new is the velocity with which business models change and new operating partnerships, with all of their attendant integration issues spring up.

It is incredibly easy to duplicate even large-scale solutions on elastic cloud infrastructures and create redundancy that may be quite difficult to corral and remediate. If there is no central model repository, or no discipline about maintaining it, it's also quite easy for the redundancy to go unrecognized until someone reconciles the cloud vendor's bill and maybe not even then.

In the end, Agile Enterprise Risk Management is about maintaining a detailed understanding of your company's anatomy from strategy down to your enablers, how the pieces and parts interrelate and which of them must be modified or evolved in order to achieve transformation.

In a future post, I will document a design for a system and repository to do exactly that.

Date Created

2020/09/24

Author

howardmwiener