Modelling: Essential Not Optional (Part 2)

(Original creator: bolarotibi)

By Ian Murphy, Principal Analyst and Bola Rotibi, Research Director, Creative Intellect Consulting

Read Part 1 here.

Complexity is inherent in our IT DNA

One of the goals of IT for decades has been to reduce the complexity of the systems it writes and maintains. There are several reasons for this. Users want solutions faster, budgets are shrinking and complexity fuels failure.

Agile development, automation, Cloud computing and DevOps are all helping IT deliver applications faster and at a lower cost. This is positive news for the business. But what about the rising issue of complexity?

Unfortunately, complexity is inherent in IT systems that are used to run businesses. Stock control needs to be integrated with sales order processing which in turn is integrated into accounting systems. Call centre teams need access to these same systems to deal with customer queries. Online shops must be able to create new customers, display stock levels, take orders, and pass data to fulfilment systems. These are just some of the very basic systems that companies use.

We are now in a mobile world where applications are now required to run in web browsers or be written for multiple operating systems and classes of devices. These devices are not owned exclusively by the business instead, they are increasingly the property of individuals.

This means that any applications deployed on the devices is not just running in the context of a controlled environment but has to coexist alongside other applications that IT has no knowledge or control over. The end result of this is an incredibly complex set of security and performance issues that IT cannot know yet has to write solutions to deal with.

A further complication is that security is a constant challenge. The rise of malware, the ability of hackers to penetrate systems, seemingly at will, the risk to corporate data and the surge of compliance requirements is seemingly never ending.

Modelling has a new relevance

There is a new relevance for modelling in IT systems. Let's take the example of an application designed to help an insurance sales team.

The requirement from the sales force is that they want an application that runs on their tablets and smartphones, that is capable of validating user details and can help deliver quotes, on the spot, that customers can sign up to.

From an IT perspective the operating system is unknown. The local storage and security capability of the devices are unknown. The application needs to integrate with customer systems which means they have to do data validation at point of entry. Information gathered needs to be risk assessed in order to create a meaningful policy and payment schedule. If there are potential problems, the application needs to be able to pass all the data to an underwriter in order to get a response.

This is just a quick list of potential issues and at every point there will be integration with other systems and the need to pass data around.

A computer model of this system might be very simple to begin with. Mobile device connecting to customer system, check for existing or new customer, data validation required, policy risk assessed and then payment schedule set.

This simple model enables key areas to be highlighted for further investigation. For example, does this have to be real-time? What performance speed is required? Can it be done over 3G or does it need a WiFI link? How long does it take to validate customer details? What happens if an underwriter is needed to make an assessment? How many users can the external gateways support at any point in time?

In short, the model encapsulates the five key points that models in general must deliver in order to be effective: In short, the problem has been **abstracted** to a mobile device connecting to core systems. **Understanding** is achieved by all parties because the abstraction is clean and contains just enough detail to see where potential problems could occur. The model is **accurate** because it describes exactly what is needed and the key steps that are involved. The identification of the external gateway as a bottleneck and the time required to carry out key tasks means that **predictions** can be done. Finally, there has been little to no **cost** at this point in establishing the model.

This is an overly simple example of a system with limited integration points but it demonstrates how quickly a model can begin to highlight areas of concern and how they can be further addressed. There would be no reason why the data validation couldn't be modelled in more detail to understand what was being gathered and how it would be validated. The same is true of the process that creates the policy and determines the payments.

Modelling: relevant and crucial for Cloud computing

One of the major impacts on the IT landscape has been the arrival of Cloud computing. Systems may exist in a private cloud, a public cloud or be split over the two in a hybrid cloud.

In all three cases there is a need to understand how an application will be architected to take advantage of the capabilities that Cloud computing offers. Six key questions surrounding any application deployment to the Cloud are:

- Where will application components sit?
- Where will data be stored?

- What is required by data protection and compliance laws?
- What level of performance and scalability does Cloud provide
- What security and encryption will be used?
- What cost savings do the different cloud models offer?

Modelling allows companies to begin to address all of these questions. At the very basic level it will show application components and highlight potential integration challenges. For data, it will enable compliance teams to determine whether the company has a legal problem. Security teams can begin to identify what is needed to meet corporate security needs.

Without a model, a lot is taken on trust and people fail to properly identify challenges. Many companies are beginning to realise that there is far more complexity in moving applications to Public and Hybrid Cloud than they would ever have realised. A model would enable them to not only see what was moving but then enable subject matter experts to ask questions about integration, security and suggest what further and detailed models are required.

Model or be damned

There is no excuse for not modelling IT systems and in particular software developments. The five stages are clear and easy to use.

The key is in keeping it simple, using models to explore potential challenges and not over complicating things. Many organisations will ultimately discover that they don't need a new model for every application and system because the similarities at the model level are very high. For example, mobile applications share a lot of common elements. Where they differ is at the accuracy and prediction stages.

Those companies that use models will identify problems sooner, reduce cost, and understand complexity. They will also open up opportunities for greater reuse and flexibility. In an age where business agility is paramount, modelling enables a company to deliver what users want, faster and with less risk.