Minimalist Information Systems Design and Architecture

Karthik Suresh

ABSTRACT

This paper looks at how we approach the design and architecture of information systems — and asks what approach would be most useful when trying to solve a problem, provide a service or improve a situation.

1. Introduction

This paper started because of a conversation I had with my son. We thought it would be a good idea to create a computer game. A very simple one. One that rolled a dice and gave you an action to do, a little like Twister or anything else that links chance and an outcome.

"We could build a big dice machine," he said. "And then type in the number and get the result."

"Do you want to build a dice machine," I said. "Or do you want to play a game?"

"Play a game," he said.

This is the point for anything we create — anything that requires design or architecture. We're trying to get something done and all the work in the middle is designed to make that happen.

2. Design thinking versus outcome thinking

Many of us are visual thinkers — we respond strongly to the aesthetics of things. It matters how things look, whether they're clean, whether they look good and if things are well managed or not. We cannot always define what good looks like but we know it when we see it.

The problem with an emphasis on the appearance of things is that we can confuse how something looks with how it works. This is perhaps most apparent in website design, where a huge amount of effort goes into making something look good but then the content is pasted in, shoehorned in to fit the image and appearance of the pages. It's even worse with marketing emails, which are so full of fluff and images that any meaningful content is lost.

The purpose of web pages and emails is to communicate with prospects and customers, to explain what you do and help them understand why they should work with you. Good design should be invisible, helping you get the point across using as many senses as possible that reinforce and communicate one key message.

Any designer will tell you, however, that's obvious. You need to know what you want to say — and then they will help you select the best way to show your message. The fault is with the customer who hopes that good design will overcome a lack of clarity, and this is focusing on building a system without thinking about what the system needs to do. Building dice machines and missing the point of the game.

3. The recipe book model

One way to approach this is to think in terms of the recipe book model. How do people write recipe books? In general, they tend to work through large categories of food types: meat, vegetables, desserts or write about different kinds of meals: breakfast, light lunches, hearty dinners. As a format, this lends itself to between two and four pages for each recipe, including a picture of the finished dish. People tend to buy books because of how they look, because of how tasty the dishes in them look. But then how do we use them?

We all have recipe books, probably a shelf full of them. There is a power law in operation here. You use a few of the books much of the time and, in those books, you use some of the recipes most of the time. When you think about it how many of the meals you make tend to be very similar week to week and month to month? If you're

time poor and too busy to cook for hours after you come home from work the chances are that you default to a small number of choices that work for you most of the time or you reach for the take-away menu if you just can't be bothered.

One part of the recipe book that tries to help out is the index. Let's say you come home and look in the fridge and all you can find is some leftover chicken and a selection of root vegetables, the index will let you look at recipes that use those ingredients and perhaps give you some inspiration on what to cook.

A recipe book, on the whole, gives you the information you need to get on and make something. The online experience of recipes is, however, turning into a new kind of hell. First, almost any website you go to seems to be designed more for search engine optimisation than for users. I suppose that makes sense, sort of, although you would hope that the search engines would serve up pages with quality content regardless. Then the pages try to either be cute and clever, requiring you to toggle between the ingredients and recipe, or they spread the information over several pages requiring you to click and interact or they intersperse the content with lots of pictures that are supposed to help but that just make it harder to follow what needs doing.

Then again, no designer can cater for everyone. Some people want precise instructions, weight down to grams and time down to minutes, so they can follow everything exactly and expect to get a perfect result. Others are more comfortable with uncertainty, with a "pinch of this" and a "simmer till done" sort of approach. So how would an information systems architect deal with the different needs and many options for presenting information?

4. The constraints that come with structure

The first thing we need to do is understand the nature of information and the different reasons why we do things with it. Information is a strange thing. In association with its cousin, knowledge, it can be created from nothing and shared without losing anything. If I think of a way to do something and I tell you about it, I haven't lost my ability to do it but you've gained some or maybe all of what I knew. Intellectual property, as a construct, is a man-made creation that tries to limit the sharing of knowledge so that the people who come up with it can make money. The way we create information or knowledge and

the ways in which we protect it, through copyright and technical means, are to do with control, and not with the underlying nature of the information itself.

As far as a user is concerned, they have no interest in the way in which you manage and control information — they're interested in the way your information helps them to do something better. When it comes to recipes, for example, they might want information based on how it fits with the diet they are on at the moment, the season they are in and the kid of weather they're experiencing or based on just what happens to be in the fridge right now. How can you cater for these different situations?

The way to think about it is to think about water, which can take any form you want when you put it in the right shape of container. Water in a glass is there for you to drink. Water in a shower lets you get clean. Water in a bath lets you luxuriate. However much you try you aren't going to be able to luxuriate in a glass of water. In the same way when you're looking for a recipe book that will help you with the diet you're on a book written to prepare hearty family meals is unlikely to hit the spot. And when it's summer and roasting outside creating a stew is probably the last thing you want to do — instead you're looking at salads and trying to figure out how to make them more appetising or how you can convince the kids to give a plate of green stuff a go.

The answer is probably to stop confusing the structure that information is provided in with the information itself. A good information system doesn't focus on structure — it focuses on helping the user get what they want done and that means it needs to be as changeable and reformable as it need to be, flexible enough to cope with different demands and change its appearance when required for a new purpose.

This is easier to say than do as putting any kind of structure around information starts to fossilise it, starts to build up a structure debt that prevents you from making radical new choices in the future. If you write down words in notebook or in a computer file, the accretion of words in a linear form makes it harder and harder to move things around later. If you've got the information in a database the decisions you make around columns and fields and the kind of information you put in there starts to restrict your options as you build a system. Structure can help you create something where there was nothing before, just like

structural steel work can help you build a house but once you put up the steel work for a bungalow your options for now building a skyscraper have reduced considerably.

5. A recipe remix

The model we need is already here, and it's called the Internet. The Internet is really a massive collection of index cards, connected to other cards that are relevant in some way. The reason the Internet is so incredibly useful is because we can create information as self contained little chunks.

You can write a recipe on a card, for example, with the ingredients and instructions and put it in a box, or in a web page. On a different page you can link to all the recipes that use chicken. On a different page you can link to recipes that have to do with seasons and on yet another you can link to the ones that have to do with a diet. This linking and cross linking of information creates something new, an information architecture that grows and expands to meet the needs of users. It's incredibly simple really, but it's hard for people that are used to the linear structure of a book or the control issues associated with intellectual property to deal with.

In some ways the ability to link related information is as important, perhaps more important, than the ability to create information these days. It's sometimes said that everything that needs to be said has been said, someone somewhere has thought about the things you're thinking about and written down what needs to be done. Your problem is to find that material. We could spend our lives forever creating new material or instead spend time finding what we need to do.

But, that isn't enough. What you find was created at a certain time for certain people and it worked for them. You might find that you need to adapt what you find, change it in some way to make it work better for you. For example, you might find a great low carb recipe but because you're intolerant to nuts you need to modify the recipe so it works for your situation. That's a remix task — where you look at what you have and change it to become what you need.

6. Conclusion

It seems to me that the purpose of information, the purpose of any system is to help someone get something done. That end result is the most important thing. We can build a big system to help make this happen, where every part of the system needs to work perfectly to get the result we want. Or we can build the simplest, smallest system we can build to get started.

The first approach tends to lead to a focus on what we're building rather than what we're helping get done. The focus, very naturally, shifts from the user to the act of creation and in doing so we often build something that is technically interesting and competent but that doesn't actually help the user very much. We end up building a very complicated dice machine when what we really need is a game.

Instead, we should probably just play the game using dice, if that's what's needed. The time to think about a system is when you need a 13-sided dice for your game and it's much easier to fire up a computer and use some code to create a random number generator than to figure out how to create a 13-sided object.

A rule to follow, then, might be to create something only when you need to and when you create something create the smallest possible implementation you can. The power of your system will come, eventually, when you connect together lots of small things that help you do something you never thought was possible.

That's when your information system that houses your recipe collection helps you make food whatever the season, and whatever you have in the fridge. More importantly, it helps you lose weight, feel better and create the family meal experiences you always wished you had.

About the author

Karthik Suresh is a Management Consultant who helps customers with energy, utility, sustainability, research, innovation and knowledge management projects. His experience includes working with large and small organisations to select and implement strategic decision systems, improve and develop management capability and deploy risk management, IT, communications and information systems projects. Please feel free to connect on LinkedIn or read more at www.karthik-suresh.com