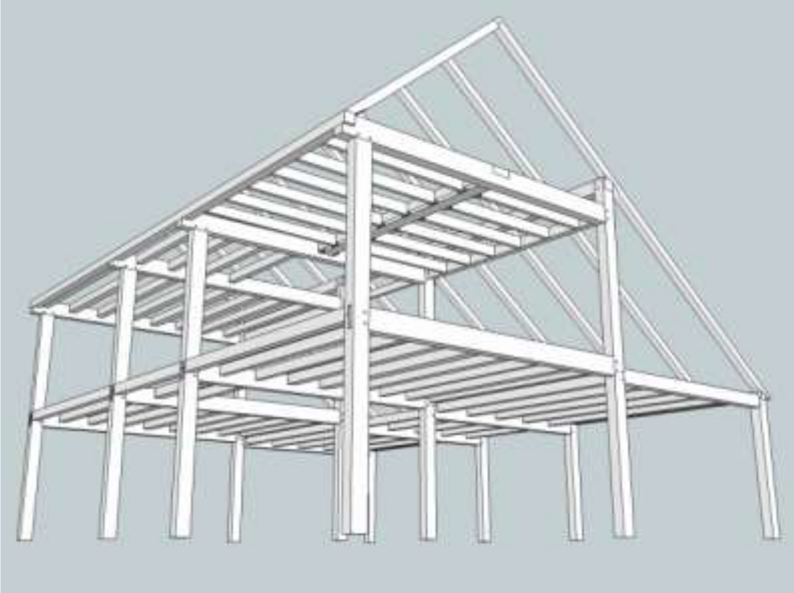
Service Management and what really matters

NOTE: You will be able to find this, and all my previous articles at www.sm4all.com.au/history

*This is the **fourth and final** in a series of four postings on some new frameworks and methodologies around Service Management. How do they work together, and what really matters?*

The frameworks under discussion:

1. ITIL 4
2. The new DevOps and Agile Service Management
3. Site Reliability Engineering
4. **What really matters**



Frameworks, Shmameworks, get over it!

For Service Management we have the following frameworks and methodologies:

- ITIL
- Lean
- ISO20000
- CMMI
- DevOps
- SRE
- ITOM
- M_o_R
- Agile
- CoBit
- Togaf
- And more

And there are many zealots of each, who will tell you why their favourite is 'the One' that works in all cases. (one ring to bind them!)

But does any of that really matter?

Do we jump on the bandwagon of the next great thing? Or is it time to go back to basics?

What really matters

In my humble opinion, there are two things that really matter:

1. **You**, and the skills, knowledge and experience you can bring to bare for the benefit of your client, and
2. **Your Client**, and the value they can derive from your services

You

What is your 'value statement'? what can you offer now, and into the future? You may be an expert in one of the frameworks or methodologies listed above. Is that good enough? Do you really think that every client issue can be fixed by your expertise, and that that is the only way it can be addressed?

Future-proof yourself by broadening your knowledge. Become a 'T-type' person. This means that you have depth in one area, and have foundation knowledge of others. Know where the other frameworks kick-in/overlap. Have the ability to apply more than one perspective to a problem.

If you an ITIL expert (for example), I would recommend you include foundation level understanding of DevOps, Agile, Testing, Security and Lean, as a minimum. As each of these disciplines are expanding (see previous postings in the series). There is greater overlap in the areas they address. ITIL expands into non-IT. DevOps, into all things IT, Agile applicable everywhere. The overlaps increase. You need to be able to appreciate these overlaps. So as to draw from several where needed.

As an example, if the client's issue is that they are not getting the customer feedback, you can draw from ITIL, DevOps, Agile, Lean and others. Each has an approach to such an issue. Offer your client the versatility to look at options.

And now the 'T-type' person should be looking at becoming a 'π-type' person. Have a backup area of deep knowledge. Expertise, especially very technical knowledge, can become quickly out of date. I still consider myself to be an expert programmer in Cobol. Not a great demand for this skill anymore.

This approach, of being a 'π-type', can future-proof yourself. When the next big thing comes along, you need only study the basics, and relate it to your current knowledge. This is how I took on the Sire Reliability Engineering role. It is very technical. I will not be an SRE, but I do know what they are expected to do, when best to use them and how they can be expected to employ Service Management (ITIL) practices.

Your Client

They have a business to run. They do not have the time, or interest, to invest in coming up to speed with the various methodologies. How would they know which one is right for them? Will it address all of their issues, now and in the future? Have you ever seen an organisation trying to keep up with the latest, only to confuse their staff by constantly changing direction and strategy? It does not work for them. It cannot work for them. That is where you come in.

The client requires your services because they have a service-related issue. Begin by understanding what that issue is. Do NOT tell them how ITIL, or DevOps will fix it. Also, beware of a client that says, "We need to implement ITIL." They probably don't.

Collect information about:

- the issue,
- services impacted,
- how it impacts them,
- what they would like to see happen, and
- what they would like you to do

Speak in their terms. Be tool and framework agnostic. Explain back to them what you understand the situation to be. Once you and your client agree on this, then start talking about possible approaches. Talk at a practice level or lower.

You are not at a solution stage yet.

Use the practice within your expertise. See if this feels right for them. Think about the other methodologies that could cover the same ground. Can you add to the perspective so the client feels they have options? Your job is to know the different options and offer them as means to address their issue.

See what sits best with them. Determine the timeframe and goals to be achieved. What fits best, small increments or big-bang? Let the client talk and you listen. I have seen consultants engaged where they wanted to convince the client how clever they were. The client wanted to escape the meeting.

Build a plan of attack. Work it through with them. Does this fit with their business / service strategy? If not, adjust accordingly. Beware if you ever catch yourself saying, "According to ITIL…".

For new clients, both you and they are not sure of the ultimate value outcome. Take an Agile approach of delivering something small, so they can try it, and give you feedback. Then plan the next steps together.

It will take longer this way. But you will get greater customer buy-in, build your own creditability, and find the best solution for their problem.

All this without mentioning any particular framework. Understanding the way frameworks interrelate is your job not theirs.

Remember, the two things that matter are You and Your Client – NOT this framework or that methodology. Be truly "Customer Focused". Always look for the best way you can assist them to solve their issues. This means keeping yourself relevant, and their situation upper most in your objective.

GOOD LUCK.