INSIDE TRACK
*Analyst commentary with a real-world edge*

FREEFORM DYNAMICS

# Improving Systems Agility

## Every little helps

By Dale Vile, August 2010

**In a nutshell:** One of the most common complaints heard from business people about the way IT serves their needs is everything always seems to take too long and cost too much to implement. This highlights the need for and value of flexibility and responsiveness, which in turn drives consideration of how computer systems can be made more agile. In order to address this, there is a need to understand historical constraints, and move forward in a manageable manner.

## Key points:

- IT vendors and large industry analyst firms often paint agile computing visions that are idealistic and essentially unachievable for the majority of organisations.
- Yet more selective and manageable adoption of virtualisation, SOA and alternative sourcing options can make a tangible difference to service levels and performance.
- Key to success, however, is an effective service oriented approach to operations and management that puts the focus on services rather than systems.
- The bottom line is that significant benefits can be realised, provided organisations set their own agenda and don’t get too distracted by vendor and analyst theory and hype.

What do men’s health and fitness magazines, IT vendor presentations, and advice from big industry analyst firms have in common? Well, apart from trying to convince you to spend money the latest stuff, they all tend to paint visions of perfection and assume you will aspire to reach them.

In the case of the magazine, it’s the ideal guy with the muscular physique, the obligatory ‘six pack’ and an ability to run marathons at the drop of a hat, while still having time to groom himself to achieve just the right amount of designer stubble. With marketing messages and analyst advice, it's the vision of an ideal IT department working in complete harmony with the business, not just delivering what users need exactly when they need it, but also actually creating business opportunities from the smart use of technology, with everything operated in a slick and cost-effective manner.

Back in the real world, of course, the number of us with any chance at all of even getting close to this kind of perfection in either context is actually very limited. The law of diminishing returns means only the most obsessed will make it, and such obsession in itself is arguably not that sensible or healthy. There are then the practical constraints which mean that few of us as individuals have time to spend four hours in the gym every day so our bodies match the magazine cover, or the luxury of putting the business on hold for a year while we redesign and re-implement everything to do with IT to emulate that elegant picture on the PowerPoint slide.

It's therefore a case of striking a balance and defining goals that are a bit more achievable. You’re likely to have a lot more success, for example, if your aim is to drop one or two waist sizes so you can fit into some of your old trousers, or get fit enough to run up the stairs in the office without gasping for breath so you can impress the girl you like on the second floor. You might not have the perfect body at the end of it, but you will certainly have achieved some things that make a positive and welcome difference to your life.

And so it is with IT. We have conducted many studies at Freeform Dynamics over the years (see www.freeformdynamics.com) confirming time and time again that while IT departments are generally far from perfect in the way they operate, a lot can be achieved through incremental improvements to policy, process, infrastructure and tooling, without replacing or overhauling the whole environment[1,2]. This is why it is such a shame that when faced with idealistic visions that are perceived to be unattainable, many conclude that discussions around improving performance and service levels using some of the approaches described don't apply to them.

## Limbering up, the manageable way

One of the areas in which IT vendors and large analyst firms have sometimes failed to connect with IT professionals in the mainstream is that of agile computing, a.k.a. the agile data centre, dynamic IT, dynamic infrastructure, flexible IT delivery, and other similar phrases; even the infamous term cloud computing crops up in this context. What we are talking about here, whichever terminology you use, is the creation of a more flexible and responsive IT environment that is better able to keep up with ever changing business requirements and the rapid pace of technology evolution.

It's a highly relevant discussion as many of the ways in which business people perceive IT to be falling short - high costs, lack of responsiveness, etc[1,2] – come about as a function of systems rigidity and service inflexibility – which are obviously the opposite of flexibility and agility.

But when you listen to all the theory and the pitches, while it all sounds very compelling, it also sounds very hard to pull off. Looking around at the mishmash of infrastructure housed in your data centre or computer room, and thinking of everything else on your plate that needs to be done by yesterday, it's hard to envisage even starting to climb the mountain towards the pinnacle of agile computing capability, let alone reaching the top. There is a dual problem of knowing how to set sensible objectives and figuring out where to start in order to achieve them.

The trick is to think of agile computing (we'll stick with this term for simplicity) as more of a direction rather than destination. It's then less a case of doing lots of new things, and more about making sure the things we would be doing anyway are adjusted as necessary to take us along the same path rather than pulling us all over the place. There are constraints that might need to be removed to achieve this, that would naturally push activity down inappropriate routes, and will discuss these as we look at some of the practicalities.

Before getting into that, however, it is worth taking some time to make sure we understand the nature of the problem that agile computing is looking to solve.

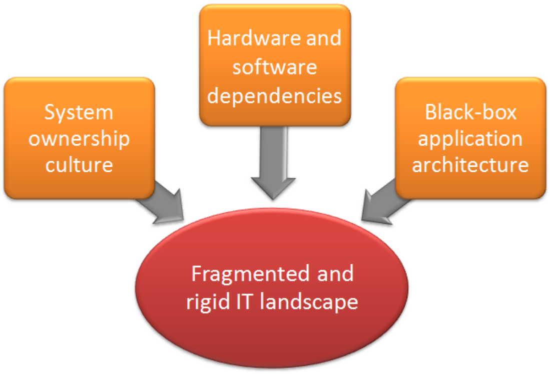
## Acknowledging historical shortcomings

The way in which we have historically implemented IT has typically led to quite a bit of rigidity and fragmentation in the systems landscape.

As a result of technical dependencies and the tight coupling between software and hardware, the commonly encountered 'black box' nature of software applications, and issues around culture, politics and accounting conventions that get in the way of infrastructure sharing[3,4], the most frequent response to a new business need has been to build or buy a new solution to meet it.

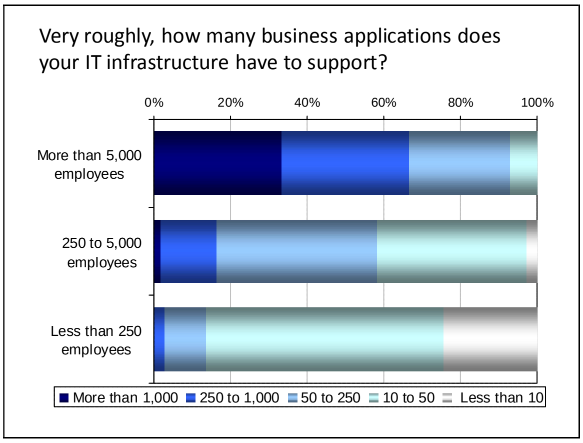
We generalise here, of course, but most people who have been in IT for any length of time will be familiar with the sentiment that it is often quicker and less hassle implement something that has been specifically designed for the job and is powered by dedicated hardware, than incurring the distraction, overhead, and sometimes compromise of trying to extend or modify an existing system, or even reuse existing kit that has spare capacity if we are talking about hardware.

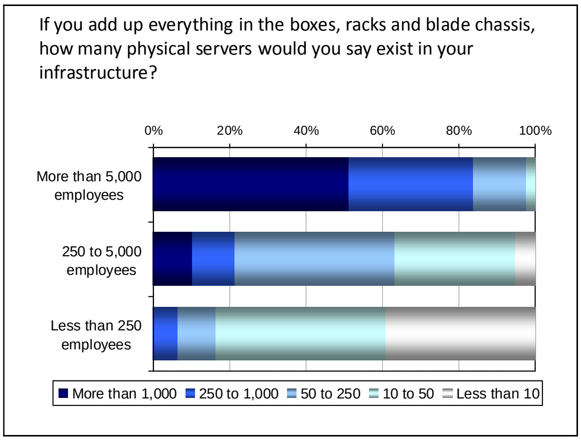
This is an area we have explored in depth in previous reports and studies[3,4,5], but for the purposes of our discussion here we can summarise the situation with the following graphic:


System ownership culture
Hardware and software dependencies
Black-box application architecture
Fragmented and rigid IT landscape

The systems ownership culture refers to the engrained assumption that business managers will ‘own’ everything that is associated with the systems used within their department or domain, including both hardware and software. Whether this is as a result of the physical hardware and software dependencies we see next, e.g. the historical need for an application to run on dedicated kit that has been configured especially for it, or more a factor of culture and accounting practices, is debatable, but certainly the two behaviours go hand in hand. There’s then the black box nature of applications, by which we mean business software that has been designed to be self contained and not to be tampered with too much, that further encourages fragmentation and redundancy.

Put all of these factors together and the end result, particularly when we consider systems based on commodity platforms such as x86 Windows or Linux servers, has been the proliferation of silos of automation, with organisations having tens or hundreds of applications each sitting on their own dedicated hardware[4].


Very roughly, how many business applications does your IT infrastructure have to support?


If you add up everything in the boxes, racks and blade chassis, how many physical servers would you say exist in your infrastructure?

In practice, of course, many of these applications do actually need to work together, and this is how we end up with rigidity, caused by endless point-to-point integration which essentially hardwires everything together in a manner that is difficult, i.e. costly and time-consuming, to change.

The other obvious implication of this whole picture is the overhead it places on IT[2,4,5]. It's not just that each new application requirement involves all of the expense, time and effort associated with procuring, installing and configuring an entire systems stack, it's the manpower and other resources necessary to keep a highly fragmented and inefficient landscape running reliably. There is then the complexity, cost and risk of dealing with functional redundancy, e.g. multiple applications doing essentially the same or similar things in subtly (or sometimes not so subtly) different ways, multiple databases holding equivalent information, and so on.

In terms of business impact, it's all of this that leads to business people often complaining that IT solutions or systems modifications always take too long to implement and cost too much to deliver.

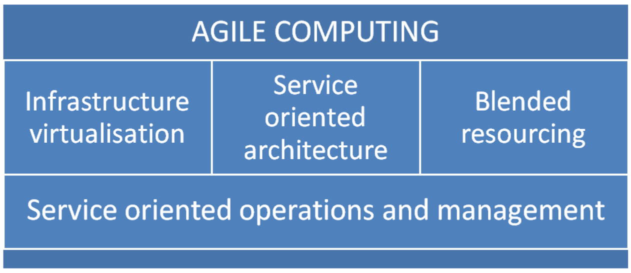
But let’s not get too carried away with over-stating the problem. Some business people, for example, have unrealistic expectations and the need is as much to educate them on the realities of IT as it is to make things better *per se*. We also mustn’t lose sight of the fact that IT service delivery is far from being a disaster in most organisations. A better way of describing the situation is that IT departments are generally doing OK, but complaints from the business camp often reflect legitimate concerns and indicate significant scope improving levels of service[1,2].

With this in mind, let's take a look at some of the enablers of agility that might help with this.

## Enablers of agility

There are various ways of considering what is needed to enable a more agile approach to IT service delivery, and some very sophisticated methodologies and maturity models have been designed by vendors, analysts and professional services organisations to help with this.

It is sometimes useful to cut through all of that complexity, however, particularly as some of it is arguably designed to create a perceived need for new technology and or consulting services, and view things a little more simply. If we do this, we end up with three main pillars of enabling principles and capability, sitting on a solid operational foundation:


AGILE COMPUTING
Infrastructure virtualisation
Service oriented architecture
Blended resourcing
Service oriented operations and management

It is beyond the scope of this paper explore each of the elements shown on this diagram in depth, but virtualisation and service oriented architecture (SOA) are likely to be familiar, at least at a high level, to those for whom they are most relevant. Having said this, if they are areas in which you have yet to get up to speed, we would encourage you to download our more comprehensive primer entitled “Evolution of Dynamic IT”[3] , which will provide the necessary grounding. That primer also covers the third pillar, blended resourcing, which is all about using the right mix of internal and external resources to optimise service delivery, from traditional outsourced development and operations (e.g. managed services), through to emerging models around more advanced hosting approaches that many are referring to today as cloud computing.

What it is important to be clear on in the context of our discussion here, however, is what the various disciplines enable in terms of agile computing capability and how:

| *Agile computing principle* | *Main benefits driven* | *How this is achieved* |
|---|---|---|
| Infrastructure virtualisation | Improved time to benefit for new applications, more consistent service levels to the business, increased resiliency, and an overall lowering of IT costs. | Faster provisioning enabling rapid app deployment, better management of demand growth/fluctuation, plus better hardware utilisation, a lowering of complexity, and simpler administration. |
| Service oriented architecture | Improved response to demands for new or modified application functionality, improved productivity, reduction of risk, and an overall lowering of IT costs. | Easier integration and testing, leading to shorter deployment times, plus greater reuse of software, further minimising integration overhead, and reducing redundancy and conflicts. |
| Blended resourcing | Overall improved responsiveness to changing business demands, along with ongoing improvements to service levels achieved. Can potentially also lower IT costs. | Reduction of skills and resource constraints and bottlenecks, and an ability to take advantage of provider specialisms, economies of scale, and more cost effective resources. |

## Every little helps

The good news is that regardless of what the various visions, architecture diagrams, methodologies and maturity models might suggest, it is not necessary to tackle all three of these pillars at once, or indeed to implement any of the disciplines across your entire organisation or infrastructure. As per our previous discussion about unachievable visions and unrealistic goals, it is better to do something in one or more of these areas than nothing at all.

This is important from a practical perspective. While some organisations may choose to put together an overarching program of improvement coordinating activity across the various dimensions, that may not be practical in other environments for organisational or political reasons. Big strategic initiatives requiring all parts of the IT organisation to pull together may also be in conflict with resourcing priorities within the various groups and disciplines, and as we said before, few have the luxury of putting the ‘day job’ on hold to transform the way everything works.

In practice, while software architects, developers and those responsible for packaged applications look at how SOA can improve flexibility and responsiveness where it makes most sense in the software layer, those looking after the data centre might separately focus on the use of virtualisation with respect to servers, storage and even desktops[6] to drive efficiency and flexibility within selective parts the physical infrastructure. We’ll provide some practical tips on specific activities to think about a little later.

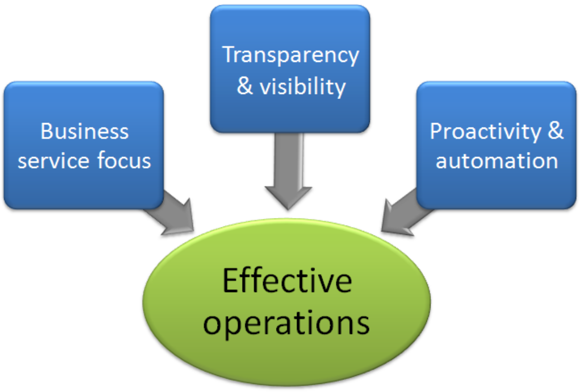
In the meantime, the one element we haven’t discussed properly yet is the foundation of service oriented operations and management.

## Service oriented operations and management

Representing the operational dimension as a foundation layer underlines how import it is to pay particular attention to it. Indeed, you can put as much virtualisation, SOA and external services into the mix as you like, but if you haven’t thought about the operational implications, you are going to hit a wall at some point in terms productivity and manageability, potentially bringing you back to where started from a sub-optimal service delivery perspective, just for a different set of reasons.

Examples here include virtual machine sprawl creating an unmanageable mess that elevates overheads, costs and risks, and ad hoc definition and ill-disciplined deployment of SOA style services that leads to performance issues and complex dependencies, getting in the way of effective troubleshooting when things don’t work as they should. And on the sourcing side of things, woe betides those who swallow the cloud computing hype about traditional IT challenges disappearing; if you implement external services without understanding all of the integration, operational and service level dependencies in the context of your infrastructure as a whole, you are asking for trouble.

All of this points to the need for some slightly different thinking with regard to operations and management. As we move away from the traditional hardwired systems stack and enable things to be changed more quickly and dynamically, things, well, change more quickly and dynamically. To deal with this, three main ingredients are required:



Again, it is beyond the scope of this paper to explore each of these in detail, though this has been done elsewhere[3]. Here, however, is a brief overview of what we mean by each:

| | |
|---|---|
| Business service focus | When software has been decoupled from hardware and SOA has disrupted the boundaries between applications, and even between internal and external capability, it becomes more difficult to define what we mean by 'a system', which in turn challenges the whole notion of 'systems management'. It is therefore necessary to look at IT from the other direction – 'business in' rather than 'infrastructure out'. The emphasis then changes from considering "is this system running as it should?" to "are users receiving the service they require?" |
| Transparency and visibility | Flexible deployment and redeployment of assets is an implicit part of agile computing. A key requirement is therefore keeping track of things – which applications are being driven by which servers, which software licences are associated with which application instances, and how it all supports specific services delivered to the business. Management tools therefore need to be more fluid in the way components are mapped and tracked, with automated 'discovery' and 'interrogation' moving from 'nice to have' to 'essential'. |
| Proactivity and automation | More rules based automation is required to drive both efficiency and responsiveness in a complex and fast moving agile computing environment, where the traditional reliance on costly and error prone manual administration is more likely to constrain performance. While the days of complete 'lights out' operation are not with us yet (and would probably make many IT professionals uncomfortable anyway), there is a middle ground defined by a mix of routine automation and 'guided' or 'assisted' administrator activity. |

Fortunately, much of the monitoring and management capability we have been discussing here is now becoming standard functionality within tools. You may even find that your current management systems already support much what's required, or can be extended relatively easily and cost effectively via additional options and modules. The challenges are typically more around adjusting mindsets and driving new ways of working.

Which brings us onto practical advice in general for steering the right course.

## Steering the right course

As discussed, it is important to consider the adoption of agile computing principles and solutions as a general direction, rather than a journey with an end point. With this in mind, here are some tips on steering the right course:

- Ensure that you as IT have an effective, open and honest dialogue and working relationship with stakeholders in the business. Unless this exists or can be created, moving forward effectively with agile computing will be significantly hampered.
- If you haven't already done so, spend time with business stakeholders to make sure the services that are expected from IT are adequately defined and prioritised. This need not be done exhaustively, indeed most people find that they have a good idea of what the critical services are already, but this provides an important foundation for planning and review.
- Perform an honest appraisal of how well IT is delivering today against business objectives and expectations. Some areas of both good and poor performance will already have been flagged up by the business, but there will be other areas in which IT knows it can do better..
- If appropriate, put a short to medium term improvement plan in place to address the most significant areas of constraint or weakness. Cost savings might form a focal point for prioritisation, but it is common to find that business people often appreciate improvements in responsiveness and alignment of IT activities much more.
- On a very specific point, if you haven't done so already, look at your hardware infrastructure and considering how you might take advantage of server and storage virtualisation in particular. Virtualisation projects in these areas tend to payback very quickly in terms of cost savings, as well as reducing the burden on IT, and increasing flexibility and responsiveness.

- If you spend a lot of time building and integrating systems in house, it may be worth training your staff on SOA principles and investing in some basic tooling and middleware. Either way, it is worth grilling packaged application suppliers on their commitment to open standards and component based architectures. For new acquisitions, it makes sense to prioritise these as selection criteria. For packages you are already using, the increased flexibility represented by SOA enablement of the latest releases may encourage you to upgrade sooner rather than later.

- Consider SOA enabled packages and other applications from a user empowerment perspective. Having a library of robust services that can be presented as widgets and add-ins for use in intranet portals and desktop tools can allow users to solve many of their own requirements through drag and drop or by clicking on a few options on a menu.

- The degree to which you explore alternative resourcing options, either professional services or on-demand hosted services, will be very much dependent on your environment and how IT operates. We would encourage you to at least consider where these might fit, however, as there is very clear evidence that open-mindedness on the sourcing front correlates strongly with successful IT service delivery[7]. The main caveat with regard to on-demand hosted services is to beware of the temptation to shortcut due diligence from a security, integration and lock-in perspective because signing up and getting going is so seductively easy.

- Tackle the challenge of traditional 'systems ownership' mindset explicitly. Changing the way in which investments are appraised and costs are accounted for isn't easy, but it's important to at least come to an understanding with the business about the need for flexibility on asset deployment and allocation. Whatever arrangement you arrive at to make this happen is secondary to the requirement for IT to have the freedom to move resources around when necessary to ensure optimum service delivery.

- One last tip based on looking at the flexibility question from a slightly different angle is to make sure you explore alternative funding options when investments are required. Options are now available from major IT vendors and their partners, plus independent financing firms who specialise in IT related matters, to work around capital constraints via arrangements ranging from traditional leasing to full project financing, including hardware, software and professional services. In an uncertain economic climate, this can really help to keep improvements flowing.

This list is clearly not exhaustive, and the mix of activities that are relevant to your environment is something you will have to work out for yourself. The import imperative is to define you own agenda in way that is manageable, and not be put off by the grand visions painted by vendors and analysts. When it comes to agile computing, it really is a case of every little helps

## References and further reading

The following research reports and papers are available for free download from the Freeform Dynamics website at www.freeformdynamics.com:

1 **Delivering Effective IT Support**
*A business management perspective*
Published August 2006
http://www.freeformdynamics.com/fullarticle.asp?aid=54

2 **Relieving the Systems Management Burden**
*No rocket science required*
Published: January 2008
http://www.freeformdynamics.com/fullarticle.asp?aid=130

3 **Evolution of Dynamic IT**
*A comprehensive primer for medium sized businesses*
Published October 2009
http://www.freeformdynamics.com/fullarticle.asp?aid=816

4 **Server Virtualization in Context**
*Rationalizing and optimizing your x86 server estate*
Published September 2009
http://www.freeformdynamics.com/fullarticle.asp?aid=789

5 **Evolution of x86 Server Estates**
*Modernisation drivers and practicalities*
Published November 2009
http://www.freeformdynamics.com/fullarticle.asp?aid=902

6 **Desktop Virtualisation**
*Early days for mainstream adoption*
Published September 2009
http://www.freeformdynamics.com/fullarticle.asp?aid=790

7 **IT on the front foot**
*Sourcing, architecture and the progressive IT organisation*
Published April 2008
http://www.freeformdynamics.com/fullarticle.asp?aid=318

## About Freeform Dynamics

Freeform Dynamics is a research and analysis firm. We track and report on the business impact of developments in the IT and communications sectors.

As part of this, we use an innovative research methodology to gather feedback directly from those involved in IT strategy, planning, procurement and implementation. Our output is therefore grounded in real-world practicality for use by mainstream business and IT professionals.

For further information or to subscribe to the Freeform Dynamics free research service, please visit www.freeformdynamics.com or contact us via info@freeformdynamics.com.

## Terms of Use

This document is provided for your general information and use only. Neither Freeform Dynamics Ltd nor any third parties provide any warranty or guarantee as to the suitability of the information provided within it for any particular purpose.