C3B Consulting's

INSIGHT-SPECTRA

Applying common sense to using technology intelligently

www.insight-spectra.com

Volume 23 Report 5; September 2010

Reflections on competing virtualization priorities and the consequences?

Charles Brett, C3B Consulting

Management introduction

Virtualization of non-mainframe class servers has been around for many years now. It has claimed many successes with substantial monetary savings as well as performance improvements and effective data center consolidation. But there is also evidence, from talking with virtualization customers and practitioners, that many (even most) organizations do not obtain the full financial and systems benefits that are achievable: they are, in effect, 'leaving save-able money in the data center'.

This analysis looks at five different aspects:

- *two different, sometimes conflicting, factors*
- *the IT approach*
- *business' desire*
- *strange consequences*
- *measure, measure, measure.*

Two different, sometimes conflicting, factors

Justifications for virtualization come thick and fast. Statistics like the following abound:

- average utilization of server processors is less than 10%
- peak time server processor utilization may be as low as 15%
- many organizations have 20-50% more computing capacity than needed
- data centers are physically and electrically bursting at the seams as more and more individually dedicated (non-mainframe) servers are added
- by 2011 more than 70% of enterprise data centers will face significant disruptions related to energy consumption and/or space availability.

Whether the objective is to consolidate, to improve management, to reduce costs or simply to be greener, there exist multiple independent (though often interlinked) arguments to justify virtualization. Yet, despite such quality justifications, often the results do not match what was intended (the one major exception being in very large enterprises where long experience with multiple mainframes and big systems almost always ensures that virtualization results are delivered as expected).

Why is there a discrepancy? Why do results not arrive? The reality would appear to be a combination of two factors:

- the differing expectations that emanate from IT organizations and from 'business managers'
- the lack of continuous, consistent and methodical measuring.

Of these, as this analysis will show, it is the latter — the lack of ongoing, accurate data — which often has the most dramatic effect. Indeed one conclusion is that virtualization will rarely deliver if there is not both the means as well as the process to provide information for analysis and modeling. A second conclusion is that virtualization is not a one-off process; it can be (and too often is) a once-off process when the alternative (of taking the longer view) will produce deeper as well as more flexible results.

The IT approach

In general IT likes virtualization. It is a shiny new toy that provides an IT organization with something fresh that looks and feels good.

Another reason that IT likes virtualization is that it can address a problem over which IT traditionally has had minimal control — the constant introduction of new servers by business units.

In the original mainframe days computers were so expensive that obtaining 80%+ utilization was an imperative: organizations could not afford to waste such a costly resource. With the arrival of first minicomputers and then i86-based (or PC-based) servers, the cost per server dropped dramatically and by as much as 2 orders of magnitude. It had become 'easy' just to buy another server; capital authorizations were

(and are) not a problem for decent servers which routinely cost significantly less than $2.5K.

The consequence of this has been that, while mainframe-class systems now exceed 95% or even 99% utilization, the average utilization — even at peak hours — of i86 type servers still remains well below 20%. In addition, the storage (plus back up and recovery) associated with each such server produces still more inefficiencies and duplications.

Furthermore, many servers are not necessarily installed in a data center. Some are in cupboards or closets or under desks and have become, therefore, points of risk exposure (they may not even be systematically managed, or known about). Others are crowding data centers, where they are at least managed by IT — but to the point where these data centers are becoming so full that their owner organizations are having to consider building additional data centers. (The ability to avoid doing this for one financial services company meant not spending $100M+ on a new data center alone, never mind the machines to populate it or the associated reduction in ongoing energy consumption).

Virtualization assists IT in the following ways:

- closet servers can be brought into a managed environment
- storage (and backup and recovery) can systematically be managed with minimal duplication (SANs, NAS, etc.)
- processor and memory utilization can rapidly improve to over 50%+ (in effect doubling utilization)
- energy consumption can be first controlled and then reduced
- space usage becomes much more efficient
- overall systems management can be concentrated and then automated in the hands of professionals who know how to keep systems going.

The problem with much of IT's approach to virtualization is that IT all too often sees virtualization as a one-off exercise (or at least one that will not have to be repeated until the next crisis looms). This attitude is perpetuated by many tools (and their vendors) that focus on the potential efficiencies that 'could be immediately realized' — rather than measuring what has been realized and could be in the future (more on this later).

Even more dispiriting is what C3B Consulting has called the 'Candy Store Effect' (see September 2009, INSIGHT-SPECTRA). This is where IT treats virtualization as an easy way not to apply common sense (for example initiating new VMs without proper authorizations, storage and backup plans, etc.).

Finally, and worse still, IT does not like to admit that it really does not know what it (and its host organization) possesses in the way of systems. While this may seem incredible, reality speaks. Almost always after intelligent discovery is run, organizations find they have more IT infrastructure (from servers to storage to components in the network) than they ever imagined. In one case an un-nameable services organization (with a top notch global reputation) found, using such an automated discovery tool, that it had 20-25% more servers in each of its 3 data centers than it believed.

Indeed, virtualization, even after its initial worthwhile 'contribution', may even become a net cost when one counts the waste of people time. This, being hard to measure, is not readily costed and, therefore, is expensively ignored.

Business' desire

Business people should adore virtualization. At its most raw, it opens doors so that virtualization:

- cuts costs
- improves IT efficiency
- increases business flexibility
- delivers a green dimension (reduced energy and space consumption).

Yet, to be fair to business people, trying to understand how virtualization works is not dissimilar to trying to comprehend why an airplane is able to fly: you need a certain degree of knowledge and awareness before you can understand. Virtualization, to the uninitiated, can seem bewildering, unbelievable and opaque — especially if the basic premise for justification (for example, that server processor utilization can be as low as 5%) is not emphasized and re-emphasized.

The missing ingredient is measurement. If the results of measurement are presented in cost-based terms ($s saved), business people immediately understand. If measurements are constant and consistent, then ongoing as well as cumulative $ savings become

identifiable. Measuring provides the proof of the pudding; the link between cost reduction and IT efficiency subsequently becomes evident to both IT and business management.

There is an additional attraction to businesses — one that sharply diverges from IT's all too common one-off approach to virtualization. Business requirements constantly change. With constant measurement, business (and IT) need to possess accurate and up to date data to enable first, relevant changes and then second, business flexibility. Virtualization should not be rigid or fixed. Virtualization should flex with the changing business requirements. But this cannot readily happen without understandable (cost-based) information.

Strange consequences

In normal circumstances virtualizing (say) 1000 servers onto a 15-25 multi-processor systems (where CPUs, memory, networks and storage are shared according to analysis of the workloads of the 1000 individual servers) would seem to deliver both business (cost) and IT (consolidation, space, efficiency, etc.) benefits. Yet one the stranger aspects of virtualization is that this may not happen — for an obvious if embarrassing reason. The issue here is whether the 'plan' actually remembers to switch off the workloads that have been virtualized …

This may seem beyond belief, Why, having done all the work to virtualize your workloads, would you not obtain the savings by switching off what is no longer required?

Most often the reason appears to be that this 'step' was forgotten, or someone decided that it would be safer to continue with the old systems. Most probably too many virtualization projects focus on the technology and the technical challenges of moving the 100 or 1000 server images from their original systems to the virtualization servers. In effect, while there may be an up-front justification for virtualization there may not be the comprehensive 'start-to-finish plan' that describes:

- the anticipated cost reductions, CAPEX and OPEX
- the steps needed to deliver virtualization
- what needs to be done once virtualization has occurred (like switching off and removing or selling the now surplus servers)
- a continuing process for measuring what has actually been saved against what was expected.

The key here is that 'start-to-finish plan'. Any decent virtualization project will produce a plan to achieve savings via virtualization and what the new approach will look like (steps 1 and 2 in the list above). The difficulty comes when the last two aspects (from the list above) are not a critical and integral part of the 'start-to-finish plan' (like making sure that superfluous servers are in fact disposed of).

Besides the obvious savings missed (like switching off the now superfluous systems), when there is not a methodical process which enables measurement of the anticipated against what actually happens, how can any organization (IT or business) know whether virtualization has achieved its objectives? Similarly, without data, how can future planning occur?

Another strange consequence of virtualization is that some of the real benefits of virtualization may not have originally been estimated or captured. For example, many (if not most) virtualizations focus on improving server process efficiency — but ignore storage, energy and space savings (the reason is simple, it may be IT seeking to solve a particular server processor or consolidation for IT efficiency issue which does not necessarily include storage and energy considerations). Yet, once virtualization has occurred, the storage, energy and space savings should be included: to ignore these is artificially to reduce the scope.

A third strange consequence derives from the all too frequent one-off nature of virtualization projects. Trying to assess further down the line what worked, is now working and is now not working is much harder without consistent analysis tools.

In an ideal world the same tool(s) that provided the initial pre-virtualization justification (preferably including all that business and IT wants to obtain) should be able to be used to assess on an on-going basis the post-virtualization environment. Unfortunately few tools do (and, because additional tools need to be bought to do this, they are often not acquired). Thus virtualized environments become a 'black box' that is only re-examined when a new pressure emerges.

Measure, measure, measure

Exploiting virtualization technology should be an

ongoing activity that delivers insights both to IT and to the business. For this to occur, measuring the pre- and well as post-virtualization environment is necessary. Here automated discovery is vital — of systems, of workloads, throughputs, usage, etc. Without this, how can any organization have the level of detail to make decisions?

Yet, for some workloads it may not make sense to virtualize initially. For example various vendors — Oracle is merely the most obvious — have decided that virtualization is an opportunity to gouge customers by demanding a huge increase in payments for software licences (to match the number of cores on a multi-way machine). This can become ridiculously expensive. By being able to use discovered data and then apply modeling it is eminently practical to shape virtualization options to optimize the blend of virtualization and 'deliberate non-virtualization' so as not to incur unnecessary licence costs.

Without the capability to measure, both the business and IT operate in a fog. This is not acceptable in today's business environment where IT efficiency needs to be assessed in terms that the business (as well as IT) can understand.

Management conclusion

Virtualization is a most useful technology. But by itself that is all it is.

Virtualization only becomes useful when an organization measures not only what happened before virtualization but what happens after. It is the analysis of data obtained that provides the continuing insights necessary to continue to obtain the full benefits that virtualization can deliver. In addition, such measurements, when combined with rigorous planning, show not only what should have happened, what did happen but what could happen. All three are important — to ensure that:

- *what was intended did occur (for example switching off now redundant servers)*
- *what actually occurred (reductions in space, energy usage as well as improvements in utilization, etc.)*
- *what further improvements are obtainable.*

In that context what must an organization look for when considering virtualization. First there is the technology (from the likes of IBM, Microsoft, Oracle, VMware and others). But this is not sufficient. To make virtualization deliver hard, proveable results, the following checklist captures what should be sought (and preferably within a single, long life tool):

- *comprehensive, agent-less, automated discovery (including against existing virtualized systems), gathered into some form of systematic inventory or store*
- *variable workload selection criteria (to include or exclude as required)*
- *virtualization of processor modeling, so different approaches and options to virtualization can be exploited and different virtualization technologies investigated*
- *virtualization of storage modeling, so different options for storage virtualization (SAN, NAS, etc.) can be exploited as well different storage architectures examined*
- *virtualization of networking modeling*
- *automated preparation of multi-year CAPEX and OPEX cost/benefit analyses*
- *automated generation of detailed start-to-finish plans (if justified by the cost/benefit analysis)*
- *the capability to continue to use and re-use the tool and all the different elements above so that new plans can be assessed, generated and executed.*

Without measurement, analyses, cost justification and start-to-finish planning, virtualization can waste much money as well as precious resources. With measurement, virtualization can reduce costs, improve efficiency, deliver a greener IT and adapt with time. Sadly, in C3B Consulting's analysis, the former is still more likely than the latter — but it does not have to stay like this.

Charles Brett
C3B Consulting
www.c3bconsulting.com

"Single contributors' anonymous" — the architect's dilemma

Larry Fulton, Consultant and Architect

Management introduction

Enterprise architecture is today at a crossroads, a point of crisis whose resolution will ultimately determine much in the future evolution of the practice, with significant implications for both business and architects.

In this analysis Larry Fulton, consultant and practising IT architect, looks at the role of the architect within and organization and how organizations do not make the most of key skills to 'cross IT/business boundaries' — at no small cost to themselves. He then draws conclusions and makes recommendations for obtaining maximum benefit.

The ongoing debate

An ongoing debate in architecture circles focuses on whether enterprise architecture is an art or a science. If it is:

- an art, then it is primarily viewed as a creative endeavor, the outcome depending largely on the skill and insight of the individual architect
- a science, then it can be studied, optimized and ultimately baked into quantifiable, repeatable methods that lead inevitably to predictable, positive business outcomes.

To be sure, the practice of enterprise architecture has evolved. In recent decades we have seen a continual emergence of ever more scientific leanings. There are:

- numerous formal, mature methodologies for describing application architecture and infrastructure; these are now suitable for organizations of any size and shape
- best practices associated with managing an organization's technology portfolio and driving compliance via technology standards
- the many activities involving certification of IT architects (as well as the nearly endless discussions about what an architect's skills, training and development should consist of).

Realities

Thus it is safe to say that a formal approach is more frequently a characteristic of mature enterprise architecture programs, or that enterprise architecture is becoming more of a science and, as such, subject to established principles and practices. Certainly, as the practice of enterprise architecture has become more formal, and best practices are shared and propagated, the influence and perceived value of enterprise architecture has increased. The fact that the majority of medium and large IT shops currently include some kind of enterprise architecture practice proves this point.

While this is good news for enterprise architecture as a whole, is it good for the architects themselves? Even more important, is it good for business?

Enterprise architecture has become increasingly effective at cataloguing the technical details of an organization's technology and application portfolios. These are areas where standard approaches, common practices and modeling tools add real value.

Enterprise architecture also continues to provide technical expertise, including addressing new technologies — covering recent evolutions like rich internet applications, cloud computing, non-relational databases and so forth — as those technologies fall within the view of corporate radar screens. Senior people with deep technical skills, whether those people are employees or consultants, have become key to satisfying these needs, and will continue to fill this role.

Yet risks exist

What is at risk, however, is whether organizations will consistently be able to match the capabilities of increasing complex technologies with the increasingly subtle and urgent needs of their business:

- where is innovation in the evolution of enterprise architecture?
- where is the stewardship of the enterprise technical capability platform — the set of

capabilities that supports the needs of the near future plus is able to evolve to meet longer-term needs?

Anyone who has spent any time as an IT architect will surely recognize some (if not all) of the following common challenges where an architect:

- is working with a project team to develop a design which overcomes the business challenges, simplifies implementation and advances the organization's technology portfolio plans — but then is excluded from key meetings where the concept is to be sold to management
- looks on as an organization whittles away the key architectural facets of its intended program that delivers critical strategic business benefits — such as flexibility and end-user productivity — through a series of uninformed, short-sighted decisions made without the ability to ensure that critical impacts are considered
- carefully evaluates available infrastructure products to identify which will most simplify the delivery of new capabilities for the business, only to have such recommendations ignored when any purchasing decision is made
- is designated an 'individual contributor' which is, within most organizations, all too often an uncomplimentary label denoting a person with 'limited perspective' and/or the stigmata of possessing an equally 'limited sphere of influence'.

These are all symptoms of architects' lack of influence where and when it counts the most — in participating in business decisions that directly affect their organization's business and technology future. While various interests are served by the current climate of exploiting the competing certifications of architects, the ongoing debates about the skills and roles of architects plus the use of the term 'architect' in an ever-increasing number of dubiously related job titles (for example, information architect, lead programmer/architect or infrastructure architect) seem pointless. The fundamental challenge is attracting sufficient attention and with this sufficuent respect for difficult to obtain and deliver skills.

This fundamental challenge is not about increasing architects' influence or job satisfaction — the stakes for businesses are much higher. With the combination of shrinking margins, aggressive re-combination and the ever-increasing complexity of IT application portfolios, improved IT structure and alignment of business and IT architecture are financial and strategic imperatives.

It is easy to fault organizational culture, senior management or merely the architects' own chain of command. Indeed the fault is almost certainly a combination of these — and the 'contribution' of many architects themselves cannot be ignored, either. Regardless: a better state of affairs for individual architects will only happen, if it ever does, if delivered by the hands of architects themselves.

So, what needs to happen?

The most important first action is that senior architects need to operate at the boundary between business and technology solutions. They need clearly to communicate across that boundary; beyond all else this should be the defining and essential skill of an effective architect. Advancement in the field of IT architecture should depend in significant measure upon demonstrating the improvement of this skill.

Business people need to understand that even as information technology, its delivery and its support are increasingly commoditized, structure and design still matter. Consider one common example: 'agility' really is only a word until the business:

- describes what it means in real business terms
- insists that architects demonstrate, in business terms, that their infrastructure and business solution proposals really do support the organization's agility objectives.

If, for example, supporting more mergers/acquisitions each year is important to the business, then surely an architecture that simplifies achieving just this should be of paramount interest to the business. If the business does not — or cannot — see this, architects need to make it their problem until the business comprehends.

Management conclusion

Architects — plus managers of architects, and people working with architects — all need to keep their eye on the prize: strategic leveraging of architects'

skills and insights to make better strategic business decisions.

As the tools of enterprise architecture mature, and as businesses look to their enterprise architecture groups to maintain and apply more information about the current environment, the 'science' facet of IT architecture must not be allowed to displace the 'art' aspects. Using all of the information available — to define a solution in terms of an enterprise architecture that delivers the maximum, sustained value that the business seeks — is what matters.

Finally, note well: if the most important activity of architects is to contribute at the boundary between business and technology solutions, then this is necessarily a job for 'individual contributors — individuals with a unique blend of valuable strategic and leadership qualities: few will have the necessary skills to operate across this boundary. As professional architects succeed at promoting theirs roles as such, perhaps the term 'individual contributor' will cease to be a pariah label and instead become a term indicating both aspiration and respect.

Larry Fulton
Consultant

The tomorrow you were promised yesterday can be here today

Peter Bye, Bye Associates/Andrew McIntyre Missenden Business Solutions

Management introduction

In today's dynamic world, organizations need IT application services which can meet new business needs and respond quickly to change. Those responsible for providing IT systems have to do so in the face of many conflicting demands, including escalating user expectations, shrinking (accelerating) time-to-market requirements and budget constraints — in essence to 'do more with less'. Although these challenges are not new, they have become increasingly urgent in the light of recent technical and economic developments.

A complicating factor in delivering new IT services is that few organizations are able to start from scratch. Most new developments must take place in the context of a variety of existing IT systems, of differing ages and technologies. These existing systems may be at the heart of an organization's business operations and processes and most larger organizations will have critical mainframe-based applications which may have been in place for an extended period and on which the business depends to serve its customers and clients. Typically such applications include core banking, mortgage processing, insurance, airline reservations, sales order processing, taxation, defense and emergency logistics.

In the light of these complications, how can an organization achieve the required degree of IT agility? In this analysis, Peter Bye and Andrew McIntyre explore the possibilities with a fresh eye.

Responding to the challenges: possible approaches

One particular concern is how to extend an existing business-critical application in order to provide new capabilities. It will often be tempting to replace it with a wholly new one, which will implement new requirements together with whatever functions continue to be needed from the existing system.

Replacement really may be necessary in some cases. One common reason is that the business model embodied in the existing application is now completely different from what that organization needs. Another is that the existing application may also be almost impossible to change. These justifications provide, perhaps, the best definition of a so-called 'legacy application': there is no reference to technology or age, just failure to fit the business model, and a perceived resistance to modification and evolution (Note 1).

But, unless it is absolutely necessary, replacement is a high risk and potentially high cost strategy, in particular if you follow what is often called the 'rip-and-replace' approach. The reasons why this approach fails so often become clearer when you look at the options available for wholesale replacement of an existing application. There are, broadly speaking, three ways of going about ripping-and-replacing:

- write a new application which retains the required business support capabilities provided by the current system yet also implements desirable new business functionality
- replace the existing application with a bought-in package
- move the existing application 'as is' to a new platform.

Developing a new application

One central problem (among many others) is how to manage the business during the time any new replacement application is being developed. The current application may be the result of many hundreds of man years of development. Unless a dramatically improved productivity tool is used for developing the new application, the project is likely to take an extended period to produce. Yet:

- how are urgent new business requirements to be delivered during this 'window of opportunity' time?
- if they really are urgent, they may have to be implemented in the existing system and of course must also be reflected in changes to the specification and requirements for the new application, thus increasing the required development effort and cost.

So the development of such a new application is likely to drag on and on, as/and requirements change. Change, of course, they will.

The alternative is that the business is unable to satisfy any new requirements during the implementation of the new application. But this can also lead to serious consequences:

- at best, the business risks losing competitiveness while it waits
- at worst it can be catastrophic if new development is seriously delayed or even fails.

The risk is that the business can be left with the current application as it was at the start of the new development, with none of the new requirements implemented. Unfortunately, IT and organizations possess far too many examples of this happening. Figure 1 illustrates the implications where:

- the horizontal axis represents time while the vertical axis represents value and cost
- the lower part shows the value delivered to the business by the current core systems

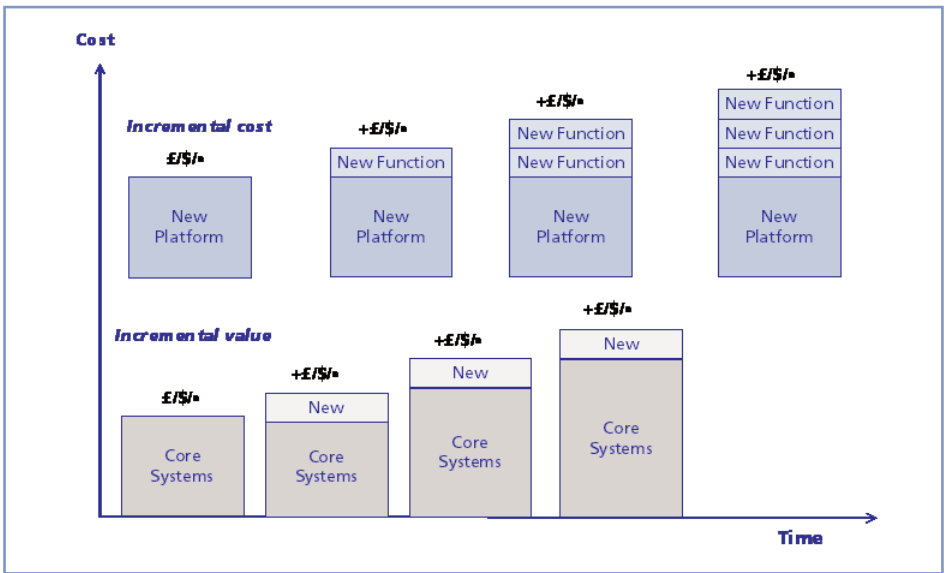

Figure 1: The implications of different styles of change

- the upper part shows the cost of implementing the new application, which is expected to deliver the same or preferably greater value.

Starting on the left, the existing core application would continue while the new one is being developed, with a switch to the new system expected at some stage. Moving to the right, we encounter a new requirement, which the business urgently needs. There is a cost to add this new requirement to the core application — but it adds value to the business. However, it must also be added to the new application, further adding to the cost of development and probably delaying the implementation.

This process can continue, extending the delay to the implementation of the new application, possibly with devastating consequences. Rarely can businesses afford to wait for new functionality which can directly contribute to revenue and profit. Yet parallel running increases dramatically with all the associated cost of duplicate operational systems.

Moving to a package

An alternative is to move to a package to replace the current application. This is a good alternative for applications that do not differentiate the business in any way.

But considerable caution is necessary for applications that are critical to deliver an organization's core business. The reasons include the following:

- a package may not deliver the required capability to support established business processes, and so these could need considerable modification (or extensions) — all of which will delay the implementation time and raise the cost
- the alternative is that the business may have to modify its business processes and the way it works to suit the package, again delaying the implementation and adding to the cost
- the problem of dealing with new business requirements that is experienced when writing a new application also arises in the case of packages
- upgrades required by package providers will need local changes to be integrated and may also provide support complications if an upgrade path is not followed.

In many cases, the implementation time and will cost far exceed original expectations and promises of the package vendor.

Moving the application 'as is' to a new platform

The third option is to move the application 'as is' to a new hardware/software platform. There are cases where such a move is possible, in particular where the application has been created using a fourth-generation or model-driven tool. Such tools provide generation mechanisms for different application environments and operating systems and may optimize the re-generation for the new platform environment.

But, in general, moving an application to a new platform requires great caution. It frequently takes much longer than expected and, in many cases, the application does not work as well in the new target environment. In addition experience shows that this approach frequently requires much more investment (in software tools as well as hardware) than anticipated.

An alternative approach

A far better approach is to develop the new functions and integrate these with the existing system. This produces an expanded as well as inherently more flexible application.

With this approach selected existing functions can progressively be moved into a new environment, (always provided there is a business case). In addition, the approach is gradual and undertaken in smaller steps, further minimizing the risks.

This is best achieved when performed within a coherent framework, or architecture, for the integration of diverse existing and new applications. One example of such a framework is SOA (Service Oriented Architecture).

Although the SOA approach will require at least some re-engineering of existing applications (to define and then expose services) experience shows that the additional effort and risk are often much lower than following one of the three replacement strategies described earlier. In addition, a logical structure (via the introduction of a services-based discipline) generates a software infrastructure which will likely last.

The reasons why

The prevalence of the adoption of the 'rip-and-replace' strategy raises a number of questions. First, why do organizations embark on such a course, when alternative much less risky approaches are possible?

There are many reasons, of which most are often less than 'business-rational', including:

- emotional reasons, which are all too frequently to do with dissatisfaction with a specific existing vendor (possibly sparked by the vendor's attitude, a significant change in pricing or even a clash of personalities at a senior level)
- the arrival of a new CIO with little or no experience of the existing systems and with strong preferences for, or experience of, something different; the new arrival may also import people from his or her earlier employment to fill other key roles in the business
- policy directives from higher authorities governing which technologies must be used; this is especially common in the public sector, where centralized authority dictates IT policy for most government purchases
- the almost blind belief that the result will both be cheaper to operate and will be quickly achieved, thereby rapidly delivering the anticipated benefits; this view is, unfortunately, heavily promoted by vendors of packages and migration services.

But perhaps the even bigger question is why so many make so many optimistic assumptions. Why are there so few quantified and cost-justified success stories of the 'rip-and- replace' approach to IT evolution?

IT is more than replete with examples of highly expensive migrations which far exceed the original cost projections and implementation timescale expectations. Even as the delays increase, projects still continue because those who authorized them in the first place do not wish to admit defeat. The fact that the private sector 'hides' many of the details of problems does not help in gathering evidence for others to use. While the public sector is more transparent and its failures known, the characteristics of these often seem so different from the private sector as to appear unreal (they are not, but their sheer size makes them seem so). Examples are legion; everyone has an array of war stories of failed rip-and-replace projects.

Management conclusion

There may indeed be reasons to move applications to 'new' locations. As practitioners it seems to us to be common sense to:

- *explore the possible consequences of each of the alternatives we have described above in full*
- *pose the hard questions about schedules and costs before starting any rip-and-replace projects.*

By using the structure described, the chances that results will be better — as do the possibilities of avoiding the waste of taxpayer or shareholder money. This must be beneficial. Yet we (the authors) see all too few organizations taking the time do the necessary basic work before starting.

Peter Bye
Bye Associates
and
Colin McIntyre
Missenden Business Solutions

Note 1: The definition of a legacy application as one that resists change was advanced by Seacord, Plakosh and Lewis of the Carnegie Mellon Software Engineering Institute in 'Modernising Legacy Systems' Addison Wesley, 2003.

Regulatory obligations may yet drive 'MDM and information fusion' to produce for customers

Nick Denning, Diegesis

Management introduction

In the May, 2010 INSIGHT-SPECTRA, Nick Denning identified reasons why organizations need to manage and exploit their information to better effect. He described a model for information fusion and considered why MDM (Master Data Management) might play an important part in delivering an information fusion capability.

In this second analysis, Mr. Denning examines what can be done and how to do it. He discusses how to obtain benefits from information fusion. Then, through the use of industry scenarios, he discusses how the business rationale for MDM and information fusion may be driven:

- *less by normal, investment-based, business justifications*
- *more by a 'game changer' — the need to satisfy legal and regulatory obligations and in so doing improve business performance (for those who understand the opportunity).*

Requirements for developing a reliable capability

The best way to avoid white elephants is to think in terms of people, process and information. This means focusing on providing the information required to support each organization's business processes. To develop this capability reliably it will be necessary to do the following:

- ensure a robust security model is in place to control access to that organization's information
- implement electronic workflows and processes which tightly integrate to electronic queries which are specific to each process and hence embed the capability in the business
- support a decision making process using information search facilities to develop queries which can then progressively be developed into policy, into practices and finally into automated processes.

What is in it for me? Obtaining buy-in from stakeholders

It is likely that the costs for deploying the range of technologies needed to achieve the above will be non-trivial. Consequently it is vital that any initiative focus on the expected benefits.

Generally we recommend a combined approach based on:

- identify a senior program sponsor
- define the strategic needs and benefits that can be delivered (by understanding and exploiting the information held within the organization)
- identify immediate points of pain
- develop capabilities that will provide a quick win
- manage actively those organizational changes that may result from improved information
- create an exploitation roadmap which enables the organization to define its technology adoption plan and the costs associated with each stage
- undertake an initial project to deliver a first quick win (in essence, providing proof of the approach and instilling confidence that the benefits to be delivered in future phases justify the investment case for those subsequent phases)
- structure the plan so that all stakeholders see early benefits (to achieve overall buy in).

There is nothing profoundly new here; this is 'just another project'. The critical issue, as always, is to create that 'eager want' within the business people.

So how does IT avoid 'pushing the next, latest technology' and instead identify the benefits for the business and the introduction phasing which will align with the organization's ability and appetite to change as it exploits the capabilities being made available? One starting point is to identify to the business a number of different benefits spread across a range of departments. In addition, each IT department should

identify other benefits specific to that organization, for example:

- for Finance: experience suggests finance teams are major beneficiaries of data fusion capabilities because they can build a reporting framework based on current, source data which in turn enables the finance team to understand the current situation as it is now, not as it was (say) three weeks ago (significantly reducing the time taken to generate accurate reports upon which the business can rely); this enables superior risk taking because of the improved feedback loop which can alert the business earlier to any issues that need addressing
- for Treasury — for understanding the likely risks involving the cash position in the business
- for Operations: an operations team can improve its decision making processes when it has access to current information; in addition it can repeat previous successes (and avoid repetition of mistakes) as well as work with the finance team to monitor key business metrics to improve forecasting
- for Sales: sales teams can improve their 'know the customer' focus, identify customer purchase patterns and plan sales initiatives to meet future needs that they expect to arise; in addition Sales can work more closely with finance teams to identify key financial criteria and plan for sales initiatives to ensure that sales activities focus on meeting key business metrics
- for Governance and Risk Management: access to up to date corporate data enables governance and risk teams to monitor the processes of the business and actively 'prowl the electronic corridors' to establish that best practices are being executed
- for the Board: provision of advanced information capabilities enables a Board to drill down into reports and then assure themselves that there is no 'corruption in the preparation of the materials'; a Board also has information that is more timely and represents the current, rather than an historic, position.

Even for IT itself there are benefits. In the storage arena it can use data fusion to reduce costs of disk space (deleting no longer relevant data), to migrate data which cannot be deleted from the production systems (to become read-only) as well as to minimize online storage volumes by eliminating duplication. In addition IT can reduce the costs of system integration, and specifically the development of flexible SOA, via integration — using MDM and developing services that present standard interfaces based on querying MDM objects. In this way IT's clients are not impacted by adding or removing systems from the MDM infrastructure. Finally, data fusion enables IT to possess a strategy which is designed to improve flexibility and reduce delivery times.

Adoption roadmap

Each organization, to reap the benefits of data fusion, needs to determine its specific approach to information management. As with any program, the overall goals need to be achieved via series of lesser individual sub-projects — each delivering benefits, the long term goals and contributing to covering the cost of infrastructure investment (something that cannot be made from an RoI on any single project).

Clarity is needed, therefore, on the current position, goals and objectives for the organization together with an assessment of the highest priority quick wins that can be achieved. On this basis, one can summarize the information fusion management elements as consisting of, and addressing, the following:

- data warehouse
- Master Data Management
- enterprise search and unstructured information mining
- a Service Orientated architecture
- data cleansing and transformation
- business intelligence and associated analytics
- information fusion's presentation.

Order, adoption — and recommendations

The issue now is how to choose the order and adoption of each technology and that this be at an appropriate pace. There are essentially two approaches you can follow.

The first is to focus on a specific need and deliver that to a smaller part of the business. In such circumstances one might implement MDM to create a 'know your customer' capability or a data warehouse for the marketing team. By focusing on a limited

capability that is specific to a single department one simplifies the organizational challenge of change — by containing the limits of change to a functional stovepipe department.

The alternative approach is to focus on improving specific processes which span multiple functional departments across an organization. In these circumstances the focus has to be on the process and the hand-offs between various internal organizations. This does not just mean automating workflow (though it may include a workflow element). Rather it means thinking about an organization's:

- decision making processes
- the information needed to support decision making
- the mechanisms for hand-off between departments (to ensure activities once started do complete).

The real implications here lie in the 'information is power game'. Experience shows that functional areas will need to make information available to the rest of the organization (which is rarely, to start with, popular). This requires a staged implementation, and probably involves at least one component of all of the above capabilities to be deployed from the beginning, even if it is only implemented to a small extent to support specific processes.

Our recommendation is that organizations should follow the second of the above for the following reasons:

- the greatest benefits will arrive with end to end business process improvements that deliver a broadly-based business benefit
- it is possible to make mistakes: purchasing software products is expensive at an enterprise scale and the second approach facilitates buying only a small number of licences initially — to support a limited process as part of a pilot (during which both the capabilities of each product and the ability to integrate them into an overall information model can develop)
- the adoption curve can utilize an implementation of a set of example processes that involve a limited number of people which then expands progressively to involve a wider and wider set of information capabilities; thus, if experience makes it necessary to change the infrastructure, the impact on users is limited until the information model has been improved (and proved)
- once the capability and flexibility are proven an organization can make the long term investment to scale up.

Another important area for an organization to consider is the set of the legal obligations that surround information management. Recent legislation — for example, the Carbon Reduction Commitment or the latest amendments to data protection laws — place significant obligations on the largest companies (and even place many commitments on all sizes of organization).

Some customer scenarios

Thus far, the discussion has been abstract, about actions and benefits and decisions. Let us add some more realistic dimensions via customer scenarios.

We see the following generic types of customer seeking information fusion:

- large corporate (including governmental) customers
- users of low volume, high value data
- users of real time monitoring, for example for command and control
- larger retailers.

We also see some potentially significant business capabilities in the following specific activities:

- targeted debt collection
- cross selling to existing customers
- understanding of a complex organization (via its data)
- treasury.

Targeted debt collection

Most organizations (whether they admit it or not) have an element of poor debt collection. The problem is often writing off money owed because this is less costly than seeking to recover small amounts.

In our experience, organizations have many databases holding the details of each small debt. If, however, an organization can cross-analyse all its systems to determine the total amount owed across all systems by any individual or organization, then the sum

of these debts for the specific individuals (or organizations) can justify recovery action. A single action improves recovery across multiple debts as well as reducing the cost of recovery per transaction.

Indeed this can be extended to recover historic debts once the total debts cross a threshold amount. This represents direct profit being added back to the bottom line, as well as valuable cash.

Cross selling to existing customers

Organizations have large databases of customer information, often in both structured and unstructured data repositories. Often there is an obligation to maintain this data for compliance purposes, especially when selling financial products.

Historically such products were sold by a specific team. Once the sale had been made, that team traditionally had little further interest (from a sales perspective). However, if an organization can provide a consolidated view of customers across all its systems, identifying cross product opportunities to sell to existing customers is much simpler. It is much cheaper to sell to existing customers than trying to acquire a brand new customer.

Understanding complex organizations

When companies acquire other companies they need quickly to understand what is occuring in an acquired organization. For acquired companies above a certain size this may be a substantial undertaking, particularly where the ethos, values and associated business practices of the two organizations are different.

The use of a combined MDM and enterprise search capability enables risk management teams to understand quickly the customers, the exposure to customers and the risks associated with each customer. More generally it facilitates comprehension of the assets and liabilities of the acquired company.

In large and highly complex companies — for example those that are found in the defense, transport or aerospace sectors and perhaps to a lesser extend in the construction industry — there are many long-lasting complex projects. A data fusion capability is essential for the project directors as well as CXOs to maintain an accurate and up-to-date understanding of the activities for which they are responsible — but without having to consume the time of project participants answering the obvious (the information can be found and assembled by the executives themselves).

Treasury

In the finance sector, products mature — and then monies have to be paid out to customers. Banks borrow short term and lend long term in order to make their margin.

All too often bank treasury departments are caught out by major cash outflows that they were not anticipating. By taking a view across all systems in a financial organization (though this also applies to corporate treasury departments, not just bank ones) it becomes simpler to anticipate when such payment outflows will take place.

Furthermore, if an organization understands its customers it can target sales teams to sell replacement products. If these new sales anticipates the expected cash outflow the effect may have a major impact on operational liquidity.

A large corporate customer scenario

The following scenario describes a large international corporate with many customers, many databases and significant regulatory obligations (such as SOX , Solvency II, FSA or SEC compliance requirements) and a substantial obligation to practise risk management. Examples might include large financial institutions, insurance companies, utility companies and telcos.

In this scenario the overriding objective is to understand customers so as to improve customer retention, to cross sell products and to assist the treasury department. An important aspect is to understand what data exists where within the business, to catalogue the information sources and then to obtain the understanding of the customers in the business.

Where the volumes of data are relatively small, even when there are a large number of systems each with complex data structures, then an MDM solution is probably the best approach to generate a quick win. Because MDM is about connecting to each repository, mapping the primary records to a master record should be relatively quick — if this activity can focus on linking the most important systems and involve only a limited amount of design for each system to be integrated. That said, success at accurately associating records to a master record may prove variable

(and require additional further work downstream). But at least initial access and the capability to search across multiple repositories will work. An example of this type of system might be for wealth management systems for high net worth individuals.

Where volumes of data are very large plus the number of systems is relatively small and the focus is on customer activity and behavior, then the focus should probably be on a data warehouse solution using:

- integration tools to populate data warehouses
- business intelligence tools to analyze data.

An example of the type of organization that might wish to follow this second type of approach is on-line gaming companies.

Dispersed low volume, high value information

In this scenario, an organization can have large amounts of information spread across multiple sources (possibly both structured and unstructured) located on multiple systems. A drain on human resources occurs because people are required to switch between applications to search for data, and that fact that data may not be accessible in the manner needed to satisfy a search.

Data in such systems is often highly sensitive; law enforcement organizations or medical records systems are typical operators of such systems. In addition, there is likely to be a significant 'intelligence' component to information management — whether to identify criminals or analyze medical trends. The issues here are, therefore, about:

- maintaining the existing security model(s)
- delivering completeness of the whole information picture
- improving staff productivity (leading to earlier resolution of enquiries within critical time periods).

This type of customer probably needs some or all of the following:

- a single sign-on capability that will enable user credentials to be defined and mapped onto the available data sources
- a capability to implement a 'flood search' across each database source in parallel (for records associated with the enquiry), but based on the user's security credentials
- provision of an enterprise search capability to analyze unstructured information (including within structured databases)
- correlation of entities detected in the unstructured information with a primary database so as to associate entities discovered in unstructured information with the primary record(s)
- implementation of an MDM capability which will enable records in all data sources to be linked to a single master record.

Real time monitoring /command and control

Organizations which have large numbers of sensors and sources often wish to be able to respond to threats as these develop (not in arrears). Examples include building management, SCADA monitoring of complex equipment, police or ambulance command and control systems, utility monitoring systems, facility control systems and even battlefield information management systems.

The issue here is that there are more data feeds than there are people to monitor them. The priority is, therefore, to provide automated real time analytics to monitor data feeds and to react to the analyses. In these circumstances the highest priority is to be able to monitor real time feeds, to detect abnormal behavior and to prioritize (to operators) the incidents as well as providing the supporting information of interest.

The priority is likely to be the implementation of unstructured information analytics to:

- process video feeds and voice feeds
- extract metadata associated with those feeds
- process these for formulating threat scenarios (such as a vehicle parked in an unauthorized location which might be a bomb threat)
- correlate that information with other engines (such as automatic number plate recognition systems to identify vehicle registrations, motor vehicle insurance and other information that can be associated with that vehicle and/or its owner).

Large retailers

Almost always the overriding requirement for large retailers is the ability to understand their customers' buying habits and the factors that affect these habits so that the retailers can ensure that they have the right products in the right position at the right time and at the right price to 'help' customers to buy. These organizations have led the way in data capture and information fusion through the use of:

- loyalty cards
- supply chain systems
- monitoring of external factors such as weather conditions
- data mining systems to identify buying patterns
- market segmentation.

Does information management work for customers?

The answer is it is doing and should do. Look at large scale retailers, for example. However, ...

It is straightforward to envision that data fusion capabilities have merit and can deliver benefit if they work as suggested. There are now an increasing number of customer case studies which demonstrate that the technologies 'do what they say on the box'.

It is, nevertheless, generally the case that these technologies are just that — technology — and not of themselves solutions. This means that the technologies require projects involving substantial effort to deliver a solution.

The critical question is whether an organization is capable of managing an information fusion project which delivers a return on investment. Without this capability, there is no point in starting.

The game changer

That said, there is a game changing factor that is coming into play — the legal obligations regarding an organization's responsibilities. These demand information fusion and management. Indeed, the extent to which an organization can satisfy its external (legal, regulatory, etc.) obligations to manage its information may well be the key to open up its abilities to improve its business performance.

If there is no option but to deploy to meet these external obligations it is common sense to address the hygiene factors of complying with regulation as well as to deliver a business benefit motivation. In other words it all comes back to defining the benefits, identifying the business case and expected RoI and then defining a program for the delivery of the required capabilities in stages to provide quick wins that justify further investment.

A challenge that we have seen is that customers often do not commit to an entire program, but rather to an individual pilot project. Such pilot projects have limited goals and successfully deliver those goals.

However, once the organization has learnt what is possible, there are conflicting pressures. Should the organization immediately extend the pilot and increase its capabilities or should it start a new program which incorporates the lessons learnt? The issue is whether to deploy the pilot into full production even though it might not comply with all organization standards, rather than to halt work and scope a replacement that fully complies with that organization's standards.

Our recommendation is that the scope of the initial pilot should include some effort to predefine the upgrade approach into full production, thereby enabling the pilot to continue to be developed in parallel with a production quality program. This approach facilitates adding new capabilities while exploiting the lessons (that continue to be learned) to be factored into a replacement design.

Quick wins

Where are you likely to see the quick wins? The characteristics of the first few projects that succeed will likely to include some or all of the following:

- demonstrating obvious, measurable benefits at the end of each project
- supporting a business process which can be deployed and exploited stand alone
- scoping of any business process that lies within the authority of the project sponsor, including the ability to direct that existing DBA teams cooperate with the project team to ensure the necessary access to the underlying systems
- focusing on exploitation of a technology with the minimum amount of bespoke code, tailoring and extension of the core capability

(with the objective of trialing all the components it is intended to deploy, and identifying any issues associated with the integration)
- indentifying critical areas specific to the business up front, and addressing these in pre-pilot proofs of concept (before significant work is undertaken that will depend on such capabilities)
- designing successive projects to minimize the number of new technologies being rolled out in any one project; this enables deliverables to be extended by adding new technology in each successive project until the whole technology architecture is deployed.

Management conclusion

The conclusions from Mr. Denning's research for these two analyses, when combined with consideration of the experience gained from talking with many organizations, make for some stark reading:

- *the benefit from any investment lies in delivering a return which requires people, process and information to be coordinated and business change to take place; the hardest part of any investment in the arena of managing the 'what's in it for me?' plus communicating what 'it' is, motivating people by creating an eager want and then delivering the associated business changes*
- *there is a significant risk that organizations will focus on building information repositories without considering the processes that need to exploit the fused information (too often IT builds these in the order that is most easy technically rather than in the order that best supports the business); in such cases the time taken to generate a return on investment is likely to be prolonged whereas as an approach that delivers quick wins is more likely to get general support*
- *there are two broad choices of approach to focus on — either specific information management capabilities that generate specific information (data warehousing, MDM or enterprise search) or process and information fusion (which starts with small deployments of all capabilities)*
- *the legal and regulatory obligations that recent general legislation places on organizations to manage data effectively are increasingly stringent (and unavoidable; specific industry requirements impose yet more obligations) all organizations are going to have to improve the management of data.*

Mr. Denning argues that both to meet legal obligations and to deliver the best overall view of data in an organization the information fusion approach will likely be the superior adoption path. By focusing on process, you need only just enough information to support the process and a minimum investment in the size of each software licence to trial and solve a light weight information problem.

Donald Rumsfeld infamously said "There are known knowns. There are things we know that we know. There are known unknowns. That is to say, there are things that we now know we don't know. But there are also unknown unknowns. These are things we do not know we don't know". The most risk lies in 'what you don't know that you don't know' though, for a commercial organization, perhaps an equally great area of risk is where 'it does not know that it knows' something. If a legal discovery process shows that certain information was within in the organization but not acted upon then this illustrates at best incompetence and at worse culpability — which can become expensive

When all is considered, management of information may yet prove to be the most important investment an organization can make to meet multiple pressures by:

- *satisfying regulatory requirements*
- *in so doing improving effectiveness, efficiency and competitiveness*
- *managing risk.*

Nick Denning
Diehesis
www.diegesis.com

Value-based case study #8: Real live issues when virtualizing 4000+ servers

Virtualization Project Manager at a Financial Services Organization

Management introduction

Mr. X is the Virtualization Project Manager at a leading organization in the financial services sector. He has many years of experience working on data and migration projects (including, more recently, virtualization ones). As he says about himself, he often is charged with diving into morasses of data to find the pearls that enable a project to proceed and succeed. Currently he is working with this international financial services organization which possesses some 7000+ Intel-based Windows and Linux servers, a large number of which his project is in the process of consolidating in order to deliver more efficient operations, to reduce power consumption as well as to record a 5-year combination of CAPEX and OPEX savings of US$150M — not least by avoiding the need to build an additional data center.

As he describes (from self-imposed anonymity), even when the 'initial' virtualization aims 'only' to consolidate some 4000 servers, the key to success is knowing what you have before you start and then being utterly methodical. In this value-based case study he discusses how his organization has approached virtualization using VMware's ESX as the base technology and Xcedex's X_Factor as the discovery, analysis, modeling and planning tool. He then describes lessons learned as well as selected best practices.

Thoughts on VMware

I am going to start with a commentary on VMware because, as will become clearer as I talk, VMware (the people not the product) is a big part of the issue. It is a global vendor. Its primary virtualization product is superb. Yet at times I wonder how it managed to create such an excellent and mature piece of technology.

Its attitude to customers is quite the opposite of its technology. I simply cannot believe how inept and uncustomer-focused it is (and I would suggest avoiding its Professional Services in particular). As a business entity you would expect it to focus on customer satisfaction through delivery of solutions; I don't. It is even worse than a professional services company that I had previously regarded as the worst.

Let me put this in another way. In the US there is a pizza chain called Little Caesar's. It has a fabulous brand image, even though most people probably have not eaten one of its pizzas (or did not like them if they did). Nevertheless, its appeal is great; it has a 'good image'. Now contrast this with VMware: the latter is the opposite. If it were not for the VMware products the company would simply fail through its lack of unconcern about customers and its apparently systematic ability to mis-scope, ignore and alienate.

Why do I say this? Virtualization — whether delivered by VMware, Microsoft, Zen or any other organization — is complex. Virtualization, if implemented well, runs at the center of an organization's IT. It consolidates many multiples of servers onto a much smaller and highly efficient infrastructure. This means that virtualization becomes a critical part of an IT infrastructure which then demands that deep and adequate preparation occur before starting down the virtualization path. Without such preparation, including thinking through how ongoing management will be delivered, virtualization runs a high risk of failing to produce the expected benefits.

The difficulty with VMware is that, despite the inherent excellence of its core ESX products, its people and approach do not seem to comprehend the gravity or significance to its customers of how its product(s) will be used. This means that any virtualizing organization will almost certainly have to turn elsewhere for planning support if it is to achieve its objectives. This is what we had to do.

Our business problem

In my experience, any time an organization says that it has 'more than 500 of anything' it is probably misleading itself about how much it actually knows. The simple truth is that when you reach such large numbers the 'real' reality makes it more likely that such a number is an estimate rather than accurate.

This was certainly true at the financial organization where I work, except that it has thousands of servers located in multiple data centers. To be able to execute

any successful virtualization initiative, the absolute key element to success is accurate information before you start. Speaking from experience, I simply cannot emphasize this enough — and yet this is not what we possessed when we started.

In this organization, for example, there exist reasonable financial records, on invoices, about what has been purchased. But invoices rarely tell you the CPU type, memory installed, disk used, network card speed and much other useful base information of a how a system is running today (though they do usually indicate serial numbers) — and certainly such financial records do not show CPU and memory utilization, peak loads, storage, I/O and network throughput, etc.

In this organization a management 'planning' exercise had occurred. This had decided that:

- 4000+ servers were candidates for virtualization in order to save data center space
- energy consumption should be reduced by 1.2MW
- server utilization should increase from an average sub-15% to well over 50%.

Deciding this was the easy part, in effect: "Virtualize 4000 servers please, out of our 9000+, and spread the revised processing load over the multiple data centers while reducing power usage and improving utilization". But which servers were to be virtualized, where and why? When I joined the virtualization team, this is what we started asking — and guess what: nobody actually knew at a detailed enough level for us really to start work.

(By the way, as an aside, do not think my organization is unique in this… and it does not only apply to IT. Think of the confusion that occurs when any company decides to let go of (say) 4000 employees. It is easy to take the big decision, but at some point detailed work is required to decide exactly which people go and who stays. In my personal view when anyone throws out large round numbers as the basis for an action plan, this is a prime indicator of some degree of basic ignorance — or a deliberate attempt to pass the buck onto someone else.)

Data, data and more data please

The big management decision had been made by the time I started. What had not been undertaken was the work which matched individual server assets and their usage to business units and locations and then recombined this into a workable plan that would achieve the given objectives. In effect the principles existed but not the detail — and very few people realized that this was the situation even though management was already counting on the savings as if, just by authorizing virtualization, the benefits magically would materialize.

Our first action, as the project team charged with delivering, was to understand the situation. At this point we took a step back. We understood that we needed to:

- obtain accurate data
- match what existed with the asset management systems
- gather the performance and throughput detail that would enable us to model, plan and then proceed.

To obtain the data to match to the existing asset management systems we knew we needed some form of automated scanning which could access all the networks and sub-networks (with appropriate authorizations, of course) and build a data store of the detail of what existed and where (CPU type, cores, memory, storage, OS, applications, etc.). In addition, in an ideal world, scanning should also be capable of obtaining utilization stats. to build the larger picture of which workloads on which servers did most (and least).

Using X_Factor

We chose to accelerate progress by using Xcedex's X_Factor as the technology to undertake scanning (what Xcedex calls 'discovery') and then to perform the analysis and modeling of candidate servers (and even modeling of specific groups of servers) as well generating a formal execution plan. Why did we choose X_Factor? The simple truth is experience: I have used or been exposed to alternative products (for example TADDM and Capacity Planner) but most of these were either too limited or too complex or simply did not gather what we needed which was:

- detailed information about each server as it was currently running (model, serial number, CPU type, cores, memory, storage, network type/speed, I/O, OS, etc.)
- sustained peak data about network usage for each server

- sustained peak data about I/O or throughput usage for each server
- sustained peak data about CPU usage for each server
- sustained peak data about memory usage for each server.

In addition, one of the post-scanning attractions of X_Factor is that it factors the last four statistics together to generate the list of candidates for virtualization (and also to identify those it is probably not worth virtualizing). It is my experience X_Factor's process/calculation here is 95%+ accurate first time in selecting those servers which are most suitable for virtualization. This saves an enormous amount of work as well as generating the accurate data store from which we could match back to the existing asset management systems (which are full, for example, of non-project relevant material like date of purchase, length of finance lease, amortization, etc.).

Planning and workloads

Another positive aspect of X_Factor is that it enables planning to occur which recognizes the realities of individual workloads and even can take into account licensing considerations. There are some workloads which it is really not practical to virtualize (in theory these can be but in practice a network or I/O or other constraint may make them poor candidates). Equally some software vendors like to extract the maximum revenue when virtualization is introduced: if you have an 8 core server, that theoretically means 8 licenses, even though only one OS image on one core might actually need that piece of software. These are considerations which X_Factor enables you to model (to include or exclude) so that unnecessary or costly virtualizations do not happen.

A further attraction of X_Factor is that it supports the data gathering, analysis and then modeling for storage and networking (not just CPU and memory, as is more common) and it can even examine existing virtual machines. These capabilities have an ongoing use in our project (as I will describe later). Our reality is that what most might think obvious for virtualization is not always a good match.

Possessing a tool that can handle inclusions as well as exclusions makes the planning more focused and less problematic. It also makes talking with the various internal support units (for example the teams supporting applications, middleware, communications, etc.) simpler and more informed: we are all talking from a position of common knowledge. In my experience about 50%-60% of potential server systems can reasonably easily be virtualized. To raise this to 80%+ (of servers) is extremely aggressive and often unrealistic once you have understood the business rules concerning different workloads.

Progress to date, including remembering to decommission

We started on this project in February 2009. We had an initial base of over 9000 Windows and Linux servers from which we had been told that about 4000 had the potential — as the apparent low hanging fruit — for early virtualization.

By September 2010 we had completed some 2500 Physical to Virtual (P2V) conversions which now run on some ~50 farms of 6 ESX servers (nowadays we are using HP blades with their density and low power usage) per farm. In addition we have decommissioned some 1800-2000 systems. We are currently doing about 120-180 P2V conversions per month.

This introduces another important dimension to our original planning. Planning is not just about moving from physical to virtual. It is also about taking out of service the original servers where the function has been moved onto a virtual farm. As part of our planning we established a detailed process that goes from end to end — with the final stage being the switching off and removal of now redundant physical servers. Remember: it is only when these old servers go that we save the space, reduce the energy consumption and derive many of the operational benefits.

It may seem surprising but it is all too easy to omit this physical decommissioning stage. In our case we tend to leave the old systems running for a period after their images have been virtualized — in case of operational issues or emergency roll-back (this has happened on only a handful of occasions). Only after (say) 2-4 weeks of this do we switch off the physical systems; after that we must arrange for these to be removed from the data centers.

You must, therefore, have a formal process to ensure that obsolete systems are removed. Otherwise you will not reap the savings. Indeed, a side benefit of a tool like X_Factor is that we can use its scanning (discovery) capability to double check that what we think should have gone has gone.

Furthermore, this scanning facility has an important continuing quality. Virtualization is not, as our timeline indicates, a one-shot activity; it is an ongoing one, and one which changes as the business requirements change. As I said, we started in 2009 and we will not finish the 4000+ until 2011. In those 3 years the business will have evolved and we must be able to flex our use of virtualization to satisfy what the business requires.

Lessons learned

My first lesson learned is 'plan, plan and plan'. Do not start until you know what you are trying to do. Make sure you have the data to be able to plan accurately and be relentless in gathering this before you start.

Scope in advance what will need to be done — not at the big picture level but at the detailed level. That way you will not run into unexpected scope surprises (or not many).

Do the basics. If we had had the data that we needed when we started in February 2009, we would have finished the 4000 P2Vs by now. Instead we had to spend several months obtaining the data, doing the analysis and modeling and then building the detailed plan with all its necessary processes (from beginning to end) before starting actual P2Vs conversions.

It is not the actual migrations that take the time but the planning. Make sure you have everything in order before you start *or* insist that you have at least 90-120 days at the start (to do the scoping, data collection and qualifying of servers which will drive the detailed planning). I cannot remember the number of times we have been told (say) 'there are 4 servers' for such a business function when our detailed stats tell us there are 7 or 9; when this happens we place that conversion on hold (and I now have 700-800 servers in this position) as we re-evaluate and re-plan to address a business need that is different to what we had been told.

In the vetting process identify your legacy systems early and know what you want to do. For example we have a cluster of 4 Win2K servers that we wanted to bring up to Windows 2003 Server. But we cannot. You cannot do a traditional P2V and upgrade from Win2K to Win2003 at the same time. The upgrade should (ideally) happen first, then the P2V. But the real difficulty is that we found there were Win2K applications that will not run on Win2003 (this has happened a dozen times so far, most usually because they were running old 16-bit applications that had been carried forward from long in the past). Not being aware of your legacy application constraints can significantly change your schedule in unintended ways.

My second lesson is to think about people resourcing. This is not so much of an issue for (say) a short term project of less than 100 P2Vs. But in something as large and long as we are doing, there is a real risk that key people will not be available when they are needed. While this cannot be eliminated entirely, it does need thought.

Indeed, in the early stages it is likely that external talent — from outside sources — will be important in being able to move forward faster, because they will have done virtualization projects before and have specialty expertise. Yet there must also be a process to ensure that this knowledge is absorbed into the host organization so that the knowledge is retained. If you do not do this you can find yourself having periods of essentially relearning how to do what has already been done. A good process to capture and manage this is essential if you are to avoid wasting time and/or slowing down when you need not.

A good way to illustrate the need for this is when, for example, your team discovers that what it next wishes to look at is on (say) an internal private network. Of itself this is not a problem — but time then passes as liaison is established and authorizations are obtained. Melding those who are internally knowledgeable with those who are technically knowledgeable saves an enormous amount of effort (the insiders would know that this was a private network in advance and so minimize delays).

These are really the primary two lessons I would highlight. Our objective has always been to create a thorough and complete process which then means that the actual P2V and subsequent decommissioning is straightforward and benign (if not always smooth).

Best practices

My first best practice is, once you have created the end to end process that I have discussed, remember there should be a transition plan for when the project finishes. We have not reached this point but I am already considering how we can hand over what has been completed in such a way that the process and

related knowledge is not lost when the primary project winds up. Unfortunately this is all too easy, especially when large organizations so readily call time and re-deploy people. The risk is that this loses hard won capabilities unless the capture of that knowledge has previously been planned for and addressed.

My second one involves the old adage 'if you don't have the metrics and I do, you cannot prove me wrong'. This is where something like X_Factor comes in great use, because it often finds and documents what others do not know (or 'know inaccurately'). That said, the real issue is about doing your preparation work thoroughly — so that you can address conflicts with better knowledge than anyone else as well as being able to do better planning than anyone else.

There is an extension to this; we have found that on several occasions our team has been brought in to explain to business units (including IT support people) what is happening with that business unit's IT usage. We have often ended up knowing more than the users (IT or business), which is a powerful tool for convincing people of what needs to be done.

My third one is that, once you have scanned and analyzed, you must ensure that all this information is used in as seamless and automated a fashion as humanly possible to run your project. You must not let disconnects occur, where information is in, or coming from, 2, 3 or 4 more different places at once.

By the time the project is finished it should be possible to say (for example), you brought us in to virtualize 4000 systems, we scanned over 7000 servers and we eliminated ~ 600 for this reason, ~ 1,000 for that reason and 200+ for a third reason. That left (say) 5,200 eligible systems. Over the course of the project we dropped another ~ 700 because of business unit considerations (or insistence) which brought us down to ~ 4,500 and here is how it all works. You should be able to trace how any one system has moved from its original server right through to the specific server farm on which it now resides — and be able to show how its performance compares with what had gone on before.

You should also be able to package up this data and take it upstream into your inventory and asset management applications. If you can do this you are enabling those who were not part of the project, and who may not fully understand what has occurred, to see what has happened and what is where. With technology like virtual machines — which are as intangible as you can find as an asset — this is important for the future. (Again, as an aside, something like X_Factor — which can scan virtual machines, and not just VMware ones — can prove incredibly useful in the future; it is not just a planning tool but can also be used as a management one.)

My final best practice is: do not regard the scanning and analysis as one-time activities. IT tends to think that you only need do it when you need something. My view is the exact opposite.

Building a data store of snapshots of what is happening in your systems provides you with long term trends, with a comparison basis for the future. In effect, by taking (say) such 5 minute snapshots, you are laying down the foundations for ongoing problem determination, evidence-based reporting and for highlighting trends as the organization evolves. This is an amazing asset to own and it constantly surprises me that so few IT organizations understand this: it is much superior than utilizing a monster CMDB-type solution which often produces so much data (captured on a per second basis) that it costs a fortune to buy, customize and maintain the tools capable of analyzing the associated data torrent. This approach is too complex. Go for something simpler and you will obtain more useful as well as usable results.

Management conclusion

It is not often that a project executive speaks out about the good and the bad. This case study is a worthy example, and it brings out how virtualization can succeed as well as what may go wrong.

Virtualization has many attractions for IT. Its attractions for businesses should be even greater as well as more focused (not least around cost and other savings). But if the right planning (with tools like X_Factor) are not used, virtualization can become an expense — rather than deliver the $150M+ of savings that are both expected and achievable in a financial services organization like this one.

INSIGHT-SPECTRA

is published and distributed worldwide by:

C3B Consulting Ltd.
19 St. Michael's Road
Winchester SO23 9JE
UK

Telephone:
+44 787 233 4000
+34 686 116 993

Email:
insight-spectra@
insight-spectra.com
or
charles.brett
@c3bconsulting.com

World Wide Web:
www.insight-spectra.com