Application Retirement: What about the Data?

Increasing the chances of migration success

Freeform Dynamics, January 2010

Introducing application retirement: what's really important to you?

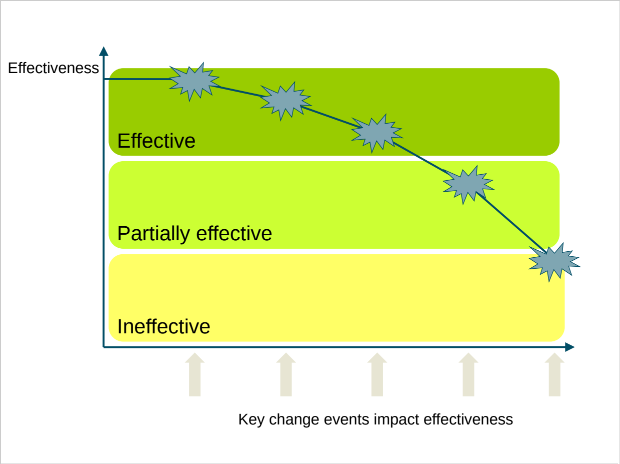
Whatever can be said about IT in today's organisations, our research shows us that we must start from the premise that 'it's not wrong'. Plentiful reasons exist why infrastructure and applications are as they are: some of them historical, because it made sense at the time, or organisational in terms of who had the budget, or indeed because yesterday's innovations can so quickly become today's legacy.

This is particularly relevant in how applications, and the repositories upon which they depend, deliver services to business users. A senior business manager once said to us:

"What's wrong with silos? At least they work. I would rather have off the shelf proprietary systems and employ four or five extra people, than spend millions on a system, and the day after its implemented, somebody comes along saying, "Can it do this?" and you end up with 3 or 4 programmers continually working on the system and modifying it."

Against this background it is unlikely we will ever arrive at what could be considered as 'application nirvana' – that is, the perfect software layer which supports the business' every requirement without ever needing to be changed. As organisations evolve and new capabilities emerge, from online banking to mobile access, so do the software and hardware platforms upon which they depend. And in turn, these impose new challenges on the data.

The downside is that IT has to deal with the consequences of constant change. We talk about such challenges as fragmentation, inefficient operations, data security and so on, which often result from where IT systems have not kept up with the changing requirements of the business. Change events can be difficult to pre-empt or indeed predict – one strategic business acquisition or change in market dynamics, such as a merger or indeed (as we have seen most recently) the credit crunch, can result in even recently deployed applications becoming more of a burden than a help.



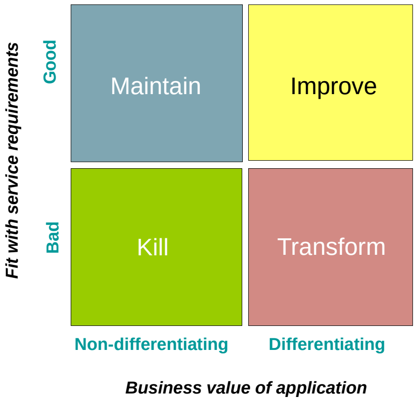
IT has to deal with constant change – but over time, effectiveness will inevitably slip.

Figure 1

And so, at various times we turn to rationalisation – that is, the act of revisiting old decisions and ensuring the existing environment remains appropriate for current and future needs. Old hands talk about technology coming and going in waves; a phase of diversification and trying new things when the going is good, will tend to be followed by a rationalisation phase when times get harder. Rationalisation kicks off with asking some fundamental questions about the relevance and currency of applications, such as:

- Is the application helping to differentiate the services offered by our business, i.e. does it contribute to business effectiveness and the top line of profitability?
- Does it help support and underpin the service levels offered, i.e. does it help minimize business costs and contribute to business efficiency?

Given the current economic environment, with capital for new projects harder to come by and operational expenditure for existing IT being closely scrutinized, it should come as little surprise that many organisations are looking to what they can to make their IT environments as efficient as possible. As shown in the figure, if the answer to both of the above questions is no, the obvious next question is, “Why is the application being used at all?”



Relevance and currency of applications boils down balancing business value with level of service provision.

Figure 2

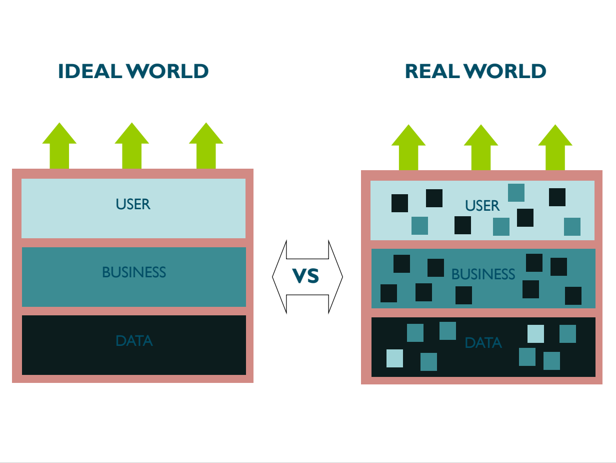
In such circumstances, application rationalisation will involve end-of-lifing older software that no longer meet the needs of the business. However and in many cases, while application functionality may no longer be relevant, the underlying data may still have value. As one senior decision maker has reminded us, "*Applications will come and applications will go, but the data we require decades from now may be the same as today*."

While there may still be value to be had from the data, it may be in as sorry state as the applications, and could also benefit from some kind of rationalisation. Indeed, not only can data access, migration and destruction issues complicate or even prevent application modernisation/ rationalisation from taking place, new applications may find that the data they rely on is a poor fit for the services they are trying to deliver.

The application rationalisation and retirement process offers an opportunity to focus on the data, not just to enable the smooth running of the exercise, but also to ensure that the resulting environment can be as efficient and effective as possible. In this paper we look at data-related aspects of rationalisation, and consider what tools and approaches might be applicable to improving both the value and cost of data.

A data-centric view on rationalisation and retirement

Let's be clear: while we might talk about applications and data like they exist independently, the two are inextricably linked. Consider: during the client server 'revolution' and before, software designers learned how important it is to separate an application into layers – the user layer, the business layer and the data layer for example – and most importantly to minimise the level of coupling between layers. However, as anybody who has tried to rationalize data or applications will know it is rarely a case that an application can be 'sliced away' from its database or repository.



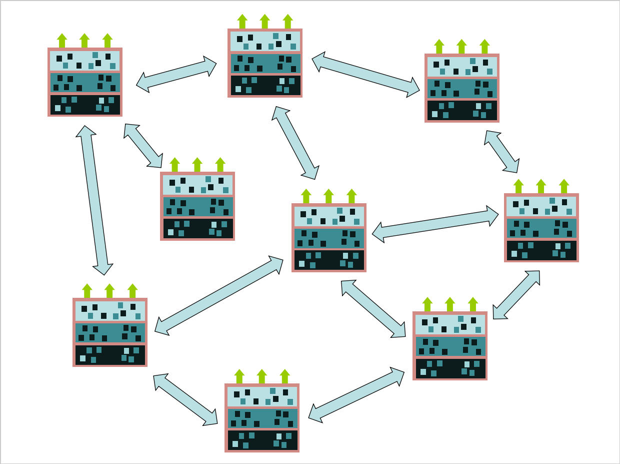
The three-layer model of application architecture is often not as neatly defined in reality – data and control may exist throughout the layers.

Figure 3

In reality however, (as shown in the figure) the three layer model is often peppered with compromise decisions and poor design choices, for example with data logic (e.g. testing data consistency and managing parallel access) being executed within the business layer of the software and indeed vice versa (e.g. business domain rules such as credit limit checks), within the data layer.

It is important not to get religious about such models: they have been hotly debated for years, if not decades, for example in terms of whether a particular bit of 'logic' is user or business logic, or where exactly control should sit. Suffice to say that if an application is to be retired and the data kept, it is not as simple as stripping off the top two layers.

To further complicate things, we need to recognise that today's IT environments are highly complex and interdependent. Again we have the 'ideal world' of software architecture and design, the best practice principles of which (e.g. Service Oriented Architecture) continue to evolve. Back in the real world once again, multiple generations of legacy software are mixed with previous integration, rationalisation and other data management activities. Middleware and batch extract-transform-load packages intermingle with lower level data protection, discovery and archiving to create a complex network of applications, repositories and supporting software.



Most larger application architectures are complex networks of software functionality.

Figure 4

It is perhaps unsurprising that half of the challenge is actually finding the data in the first place. As we know from studies around security, governance and risk, many problems arise from applications and data that have been ignored or even forgotten. To cut through the jungle we're going to have to strip things back to the simplest possible terms – and use these as a basis for some hard-nosed prioritisation about exactly what we need to fix.

Starting with the data management end in mind

Not all data is created equal. At its heart, the data an organisation holds is a manifestation of how it does business – with whom, and with what, and how, where and when. It makes sense therefore that data is considered in terms of what the business is trying to achieve, just as much, indeed, as the services applications are looking to deliver.

Any organisation should be able to identify two types of data which can be addressed as part of a rationalisation process. First, there is data the business *wants* to keep – because it is helpful to the business. Ideally the organisation will want to keep this as long as possible, but no longer (lest it becomes a burden, or risk). A simplistic view is that the value of data plummets initially and then levels off, as information is accessed gradually less often. This model can vary where analytical tools are used, and of course where there is a requirement for archiving and discovery – all businesses will be different in how they use older information.

Second, we have data the business *has to* keep: for compliance, reporting or other policy-based reasons. In this case the accessibility requirement remains constant for the period of time that the policy applies, but the business will be looking to (or may have to) delete or otherwise render inaccessible the information once the defined time is reached.

In general, a rationalisation exercise should seek to maximise the value of the data the business wants to keep, considering for example how it can be best used to deliver services and (where appropriate) increase profits. It should also be looking to minimise the costs of data the business has to keep, in part by reducing the quantity, but also by limiting associated operational costs. What's going to affect both the value and cost of data in many cases depends on the data itself.

Fundamentally, 'input' to the rationalisation process is a set of data repositories, databases and so on, all of which can in some way be treated using a range of tools, technologies and approaches.

We shall look at the options for treatment in the next section, but first, let us consider *why* we might want to treat the information. Consider the following – all of which can be treated in some way during rationalisation:

- **Data consistency and quality.** A common issue is that existing data records and/or repositories are inconsistent or even contradictory. Data quality and integrity is not a given, and it may even be for these reasons that an application is being retired. But all efforts need to take place to ensure that data quality remains stable, or improves during the retirement process. The data state needs to be assured at all times; before, during and after an application is retired.

- **Accessibility and scalability.** Existing information may well be fragmented, formatted in ways that need translation and/or duplicated across repositories, as illustrated by the elusive 'single view of the customer' that so many organizations have striven for. As well as resolving such historical issues, data will need to be rendered accessible to users via the new application or interface, and post-rationalisation the data may need to meet new requirements on access by a potentially larger number of concurrent users.

- **Security, protection and compliance.** Risks on the data need to be mitigated both during and after the migration, not least through the use of appropriate data protection mechanisms. Indeed there may be retention and other compliance-related requirements that need to be taken into account across all stages. Both during and post rationalisation, mechanisms need to ensure that business users can get on with their jobs even if something goes wrong, from systems failure to theft, fire or flood.

- **Management and operations.** A driver for rationalisation may be that the data itself is currently stored or managed sub-optimally, for example it could have outgrown, or be close to outgrowing the allocated capacity. As well as migrating to a more appropriate platform, service levels in place will need to be reviewed and confirmed with respect to whatever is planned for the data, and operational aspects such as service desk, fault diagnosis and resolution, measurement and reporting all need to be taken into account.

The rationalisation process offers an opportunity to deal with some of these issues. Remember however, it will only be worth doing if there is some positive business impact to be gained. As discussed, 'just' ensuring that the information assets are protected on their journey from an old silo to a new one would be a wasted opportunity. Instead, why not start with the idea that there is business value in having the data that emerges from the rationalisation process being slicker, cleaner, higher value than the data that goes in?

We can measure impact in terms of the 'benefits minus the costs associated with the changes'. For example; one database contains information of dubious quality, but to improve it may be expensive for only a marginal gain to the organisation. However, such gains do not only have direct financial implications, but also indirect for example:

- Enabling the slimming down of business processes that make use of the information. In blunt terms this can often refer to 'rekeying' – the archaic but still prevalent activity of reading information from one screen, and typing it in at another.

- Reducing the risks that may be associated with inconsistent or fragmented information – for example, customer records that exist in more than one place can result in inconsistent customer service.

- Minimising the costs of compliance, for example the overheads associated with 'discovery' when specific information needs to be collated. This can be very expensive indeed if information is difficult to find/access or poor quality.

- Moving towards a single version of the truth – enabling the more efficient sharing of data between applications in the first instance, and supporting higher-level initiatives such as master data management.

- And of course, building an efficient platform for data management and operations, not only to trim application and data management support costs, as well as software licensing costs, but also to support efforts around business continuity and disaster recovery.

This may sound obvious but as we all know, operational environments tend to be complex and fast-moving, so it can often be difficult to see what needs to be done (with the frequent result that 'do nothing' is often the outcome). The answer is prioritisation, in terms of the value of the information to the business, and the positive impact of making any changes, to focus the effort where it will make the most difference. We shall look at how to approach this in a moment, but first let us consider what tools we have in the tool chest.

Technology Evolution

Given the complexities around data that have grown over the years, it is unsurprising that a wide variety of tools and technologies exist to manipulate, convert, integrate, consolidate and otherwise process it. Here's a brief overview of the kinds of capabilities we have available to us.

- **Identification and analysis of data sources.** As mentioned, many organisations do not even know where all their information lives, never mind the dependencies between databases and other repositories. Tools exist to identify what information is stored where, at both a repository level and also in terms of the data itself, its business relevance and inter-relationships.

- **Archiving and retrieval.** Archiving software is used to move data from more expensive 'frontline' storage to lower cost storage without placing undue overheads on business processes and users with a need to access older data. Such tools also have relevance to discovery requirements imposed by compliance. Archived data can be indexed, rendering it searchable and retrievable. This is critical to any retirement project since an organization can no longer depend on the application to retrieve the data. The data must be accessible independent of the application.

- **Data movement and migration**. For many years, capabilities have existed to extract data from one repository, manipulate and/or transform it in some way, and import it to another repository. Data movement mechanisms can account for existing data integrity/attribute (e.g. the preservation of references or other metadata), such that it can be relied upon by the new application.

- **Data cleansing and quality management.** Structured data quality is an age-old challenge, with issues surfacing from poor data entry, to incomplete or incorrect data models. The migration process is an ideal time to deal with issues of 'dirty data' as it is moved from older repositories and before it is rendered accessible to new applications. Tools exist to automatically or semi-automatically review existing data, allowing for manual intervention as necessary.

- **De-duplication and database compression.** It is perhaps a surprise that data de-duplication technologies are a relatively recent innovation, but less surprising is that demand is growing fast given the potential to reduce the required storage space by an order of magnitude. De-duplication tends to work alongside other technologies such as backup and archiving. Database compression technologies also exist, particularly for legacy data, which can be moved to a much smaller footprint.

- **Data destruction.** In some cases it will be necessary to render data inaccessible. A number of techniques exist, including over-writing of information or indeed physically destroying the medium (disk, tape etc.) upon which the data is stored. Clearly, part of the challenge is to locate where the information has been stored, backed up or archived.

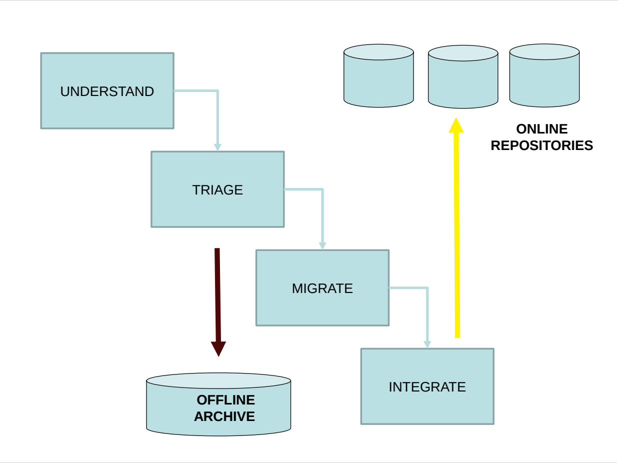
It is important to recognise the adage of, 'right tool for the job' – too often in IT, we can tend to treat data on the basis of what we have available, as opposed to what is most appropriate. Any approach needs to start from the point of view of what we need, not what we have: we look at this next.

Taking things forward

As we frequently say; IT can never be just about technology. In rationalisation it is important to adopt an approach that fits the overarching business requirement, the information assets to be dealt with, and any constraints that need to be taken into consideration. Some approaches will be more appropriate than others of course, but all will include the following key elements:

- Understanding the data and application environment
- Triaging and archiving data that does not need to be migrated
- Migrating and rationalising applicable data
- Integrating rationalised data with new applications

Let's look at these 4 elements in more detail.



Triage reduces the load on rationalisation, enabling better focus on high-priority data to be kept online and integrated with new applications.

Figure 5

Understanding the data and application environment

This first step is about building a picture of what needs to be dealt with, based on what we have already discussed – the value to the business of the data, and the costs that will be incurred to migrate, archive, cleanse or otherwise improve it.

A clear picture of the data environment will include:

- The dependencies between applications, their business users and information repositories in all their forms.
- The logical views of data – for example, what data can be considered as 'customer data' or 'product data', together with the physical locations of the data.
- Assessments of the impact of applications to be retired in terms of the both data and the business users who still require access.
- An idea of the value of the 'data under rationalisation', such that rationalisation activities can be prioritised and undertaken appropriately.

The goal here is very much to focus on the data that both (a) is relevant to the rationalisation process and (b) can benefit from some kind of treatment. This takes us to the second stage.

Triaging and archiving data that does not need to be migrated

Rationalisation requires a hefty dose of the Pareto principle – that is, identifying which subset of information is worth treating. It is equally important to identify information that is of less value, such that it can be removed from the picture.

Techniques for dealing with data at this stage include general migration, archiving and destruction. The first question is whether any data needs to be destroyed for compliance reasons – this is a pretty binary decision, so this subset of data can be removed from play however is appropriate. For the rest, it is about balancing accessibility requirements with value to the business as follows:

	Accessibility high	Accessibility low
Value to the business high	This category will require access by the business, potentially via new applications. It is therefore the high priority for migration and rationalisation activity.	Information in this category will benefit from being rendered as efficient to manage as possible, for example by being archived.
Value to the business low	Data in this category will benefit from being rendered accessible in an easy-to-manage format, for example as a compressed snapshot.	This information should be seen as the lowest priority and can be considered for lower-cost archiving, for example directly to tape.

Carrying out a triage process can help generate a data migration plan, which lists each information source to be treated and shows how, when and where treatment is to take place.

Migrating and rationalising applicable data

Once a migration plan is available, it 'simply' becomes a case of undertaking the migration activities as defined. We say 'simply' as there is never anything simple about rationalisation. To keep things moving in the right direction however, the two things to take into account are the dependencies on old and new applications/use cases, and the risks associated with rationalisation activities.

To treat both, the best advice we could give is to adopt a staged approach in which no set of dependencies and/or risks is ever so great that it could become a real challenge for the organisation were something untoward to happen. Migration of single, high-risk databases should take place when there are no other distractions and once all necessary pieces are in place (in the words of Sun Tzu; '*never fight a battle on more than one front at once*') so, for example, ensure the new application is ready to use the data prior to attempting migration.

A critical way of reducing risk here is of course, testing. You can use snapshot tools to take copies of data which can then be used in live pilots, for example. You should test not only your ability to migrate the information, but also that it is accessible as planned from new applications and interfaces – or you might find yourself with unexpected headaches, and indeed costs. Forewarned is fore-armed: Given that data is being moved from an existing, and presumably working environment, the worst possible scenario is to attempt to move it and fail, leaving business users unable to work at all.

Integrating rationalised data with new applications

When it does come to the migration itself, sufficient preparation means much of the process will revolve around user-facing activities. In particular you should be thinking about:

- Reviewing the 'to-be' application architecture and ensuring all data management requirements are covered.

- Ensuring users are sufficiently trained in the new applications, tools and access mechanisms – for example if data needs to be retrieved form an archive.

- Building in acceptance testing both to ensure that needs are met and to enable confidence building among users.

- Planning sufficient down time and ensuring it does not conflict with key business events, for example, the monthly salary run.

- Once again building in appropriate fallback mechanisms, so that users can continue to work should something go wrong.

As with any deployment, integration and testing of the modified repositories against the new application will be necessary to ensure the rationalised data is production-ready. We should not underestimate just how complicated these activities can become. Dependencies often emerge quite late on in the process. For example, it is not uncommon to discover network bandwidth is insufficient to enable migration to take place while keeping existing systems running.

The keys to migration success

Given the complexities around data rationalisation, it is perhaps no wonder that many organisations choose to do nothing rather than attempting to improve their lot. Unsurprisingly then, the rationalisation activities that take place are often those that have to, rather than those that we want to benefit the business.

Here are some final pointers to the essential elements of any application retirement strategy with respect to data management:

- It is as much about dependencies as data – between users, applications, repositories, physical and logical. Ignore these at your peril!

- Context is king – data by itself may lack the business information it requires to make sense – for example, a certain screen may show the mapping between two address fields, which would not be clear from the fields themselves. So, do ensure such contextual information is retained and accessible post-rationalisation.

- Don't boil the ocean. While this is common sense, prioritisation exercises can sometimes unravel due to scope creep. So review whether or not the current scope is achievable, and if not, fall back and regroup around the highest priority items first.

- Plan to fail – or at least, be sure that you know what to do if something goes wrong. Allow sufficient time for testing, and build in fall back paths wherever possible. Consider business continuity requirements during and following the rationalisation process.

- Get buy-in – recognise that application retirement is a change programme, so be sure to maintain a view on the business goals it is targeted to achieve and ensure that relevant stakeholders are kept in the loop.

- Take the opportunity to do things better than in the past. Rationalisation is a good opportunity to incorporate appropriate controls and management hooks – for example, even if retention policies are not yet set, you can at least ensure that retention capabilities are considered as part of the solution.

There are no absolutes, particularly in rationalisation. However there *are* things you can do to maximise the value you achieve and mitigate the risks to your business during and after. We hope this guide has been useful and we welcome any feedback you may have.

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.

About IBM

At IBM, we strive to lead in the invention, development and manufacture of the industry's most advanced information technologies, including computer systems, software, storage systems and microelectronics.

We translate these advanced technologies into value for our customers through our professional solutions, services and consulting businesses worldwide.

For more information on IBM, please visit www.ibm.com.

Terms of Use

This report is provided for your general information and use only. No warranty or guarantee is provided by Freeform Dynamics Ltd, or any third parties with respect to the suitability of the information provided within this document for any particular purpose.