



Market Review

Market Review Paper by Bloor

Author **David Norfolk**

Publish date **June 2021**

The Future of Business ...using Enterprise Server 3.0 Services

“

A company has to stay in business while modernising its systems and any modernisation must have a clearly documented business case and properly managed risk. What this means is that migrating a working system to a new platform may not be a good use of resources, especially as alternative modernisation options (such as the provision of cloud APIs or Application Programming Interfaces) are available.

”

Executive summary

The Future of Business will be largely built on the past, for existing companies.

Modernisation of what you have already, presumably “*fit for current purpose*” (or you wouldn’t be in business) avoids waste and mitigates certain classes of risk (although, if not done properly, it can add new risks). Modernisation, of course, implies fit for evolution and change – once modernised, you will be making further changes to accommodate new business.

For big enterprises, the world still runs on very large, very resilient, servers – often referred to as “*mainframes*”, although Bloor prefers the term Enterprise Server 3.0. In 2021/2, the University of Surrey is offering a **Mainframe Computing Module (COM3015)**. Promoting this module, it says that “*80% of the world’s corporate data resides or originates on IBM mainframes. IBM mainframes (“System z”) are used by 95% of US Fortune 500 companies, 71% of global Fortune 500 companies, all of the top 25 worldwide banks, 23 of the top 25 US retailers and 9 out of 10 of the world’s largest insurance companies. The mainframe operating system z/OS supports large-scale databases through DB2, IBM’s flagship mainframe database software, and IBM’s mainframe transaction serving software CICS handles more than 30 billion transactions per day.*”

And not just IBM mainframes – vendors such as Unisys still sell very large resilient Intel 8086 servers (that is, servers using the chip designs used in ordinary PCs) that fit the definition of **Enterprise Server 3.0**: a server that can “*add or hot swap system capacity without disruption... handle very high volume input and output (I/O) and emphasize throughput computing... replace dozens or even hundreds of smaller servers*”. See more about Enterprise Server 3.0 on Bloor’s website, [here](#).

Bloor believes that the Future of Business (FoB) is mutable. That is, future businesses will be in a constant state of evolutionary change, in response to rapidly evolving business environments (often, in the shorter term, responding to the consequences of the COVID 19 pandemic). So, what is the place of Enterprise Server 3.0 in the FoB?

Well, the nub of the business issue we spotlight here is that migration to Cloud is often recommended simplistically as a platform for future business, almost as a fashion option, with implied promises of ultimate agility, elastic capabilities without limits and low cost; but the actuality is often different – and never quite being able to complete a migration off Enterprise Server 3.0, because the business realities won’t let you, is probably the most expensive scenario of all. Cloud provides a wide choice of platforms, some with innovative and attractive characteristics, but three truths remain important:

- Workloads must run on the most appropriate platform for satisfying the business need, with a business case justification.
- The choice of appropriate platform must be a fact-based, not a fashion-based, decision.
- If you change platform for existing workloads, regardless of the economics of the destination platform, the economics of the journey matter.

A properly managed enterprise server has capabilities in the areas of security, resilience, performance and reliability that are hard to match with clusters of 8086 servers, running commodity operating systems. Moreover, even if an apparently more modern platform might be your preferred choice if starting from scratch today (although remember that a modern Enterprise Server 3.0 actually implements state of the art technology), migrating an existing system with proven regulatory compliance and value delivery to a new platform will be both expensive and risky; in large part because of the logistics and cost of proving that its business behaviours haven’t changed over the migration. More to the point, there may well be no obvious and holistic business benefit from the migration.

The bottom line is that a company has to stay in business while modernising its systems and any modernisation must have a clearly documented business case and properly managed risk. What this



Modernisation of what you have already, presumably “*fit for current purpose*” (or you wouldn’t be in business) avoids waste and mitigates certain classes of risk.





This paper examines the options for modernising existing business systems running on Enterprise Server 3.0 in more detail, and examines the sources of assistance that are available.



means is that migrating a working system to a new platform may not be a good use of resources, especially as alternative modernisation options (such as the provision of cloud APIs or Application Programming Interfaces) are available.

This paper examines the options for modernising existing business systems running on Enterprise Server 3.0 in more detail, and examines the sources of assistance that are available. We don't see a world of discrete technology platforms with more or less cumbersome interfaces any more than we see the Atlantic Ocean as physically separate from the Pacific and the other 4 oceans – there is one ***global ocean***. We see all the traditional technology platforms, in a well managed organisation, as an integrated whole, with currents of data flowing between the different areas, each with different local characteristics, of the technology ocean.

The Enterprise Server 3 platform has extremely powerful database services, capable of processing huge volumes of data and servicing tens of thousands (even, in special cases, over a million) transactions per second, according to ***Precisely***. And we are not just talking about DB2 relational database but the IMS hierarchical database; Adabas inverted list database and the Natural 4GL; IDMS network database; and others. This is a huge subject, however, deserving a paper in its own right, and we will not look at these further here (***Philip*** and ***Daniel Howard*** are looking at ES-3 database tools for Bloor). We will just point out that it is important to compare like with like when assessing ES-3 databases – they may not look very cost effective for small datasets but at the very largest scale, competitors may simply not bother to compete.

Similarly, we regard ES-3 security (with tools such as RACF, Top Secret and ACF2) as out of scope here. ***Fran Howarth*** covers ES-3 security for Bloor.

This report is intended for C-level management, Enterprise Architects and business managers who wish to understand the modernisation options available to them.

Modernising Enterprise Server 3.0

Future of Business – modernising Enterprise Server 3.0 applications

Bloor believes that the Future of Business lies with “mutable businesses” in a constant state of evolutionary change, in responding to rapidly evolving business environments (in the short term, this has been speeded up due to the consequences of the COVID 19 pandemic, but it has been a trend since before that). Nevertheless, some things don’t change so much, or so quickly (especially in highly regulated industries such as banking) and it is important that you don’t go out of business while navigating your journey through the changes involved in becoming a “mutable business”. What this means is that the business-critical systems of record in a large enterprise, usually running on a big powerful mainframe and often written in languages like COBOL, have to be modernised or replaced, in order to support a 21st century Mutable Business, and the strategy chosen for this modernisation/ replacement process will have profound implications for the business.

To start with, let us just review this “mainframe” concept. Forget about the 1980s room-sized monolith with its green-screen command-line interfaces. Today’s mainframe (**IBM Z** for example), although based on the same well-designed architecture, is smaller, more powerful, faster, capable of running Linux on built-in UNIX “blades” and with a variety of graphical interfaces available to it. The term “mainframe” comes with baggage that is no longer relevant, as we shall see, so (in order to avoid confusion) Bloor prefers to talk about Enterprise Server v3.0 (**ES-3**). ES-1 was the old 1980s Mainframe; ES-2 is the 1990s commodity processor cluster (although this never finally replaced ES-1 in many organisations) and ES-3 is “just another server” although a server with abilities (throughput, parallel processing, performance, scalability, potential security and resilience – you can replace processors without bringing down the business service – that the commodity competition finds it hard to match.

ES-3 is still in use, in fact, **usage is increasing** if anything. In its brochure for its 2021/2 Mainframe Computing module, the University of Surrey is offering a **Mainframe Computing Module** claims that. “80% of the world’s corporate data resides or originates on IBM mainframes. IBM mainframes (“System z”) are used by 95% of US Fortune 500 companies, 71% of global Fortune 500 companies, all of the top 25 worldwide banks, 23 of the top 25 US retailers and 9 out of 10 of the world’s largest insurance companies. The mainframe operating system z/OS supports large-scale databases through DB2, IBM’s flagship mainframe database software, and IBM’s mainframe transaction serving software CICS handles more than 30 billion transactions per day”.

And it is not just IBM mainframes – vendors such as Unisys still sell very large resilient Intel 8086 servers (that is, servers using the chip designs used in ordinary PCs) that fit the **definition** of Enterprise Server 3.0: a server that can “add or hot swap system capacity without disruption... handle very high volume input and output (I/O) and emphasize throughput computing... replace dozens or even hundreds of smaller servers”.

We also need to define the term Modernisation. Following Rocket Software and others, we see it as “investing to innovate”, so that ES-3 systems not only support the status quo efficiently and cost-effectively but also support new use cases as just another automation service. Modernisation may sometimes involve re-platforming, but the goal is to always be running workloads on the most appropriate platform and in many cases, this will still be ES-3. In fact, after investigating modernisation of IBM Z or i systems in 440 businesses worldwide, **IDC found that:**

“businesses that remain on what are sometimes referred to as “legacy” platforms and that take advantage of the plethora of hardware and software innovations that have been made available for those platforms have an overall better outcome, quantitatively and qualitatively, than those that move off them”.

“

Definition of Enterprise Server 3.0: a server that can “add or hot swap system capacity without disruption... handle very high volume input and output (I/O) and emphasize throughput computing... replace dozens or even hundreds of smaller servers”.

”



There must always be a business case, based on balancing Value and Cost, for making the choice between reusing, modernising, re-platforming and replacing legacy ES-3 applications.



What this means is that there must always be a business case, based on balancing Value and Cost, for making the choice between reusing, modernising, re-platforming and replacing legacy ES-3 applications. Just because “everyone knows” that “the mainframe is expensive and cumbersome”, you can’t skip the holistic lifetime business case and shift everything onto distributed clusters, because what “everyone knows” is often wrong or, at best, out of date.

Dropping down a level, as IBM has always said about its Z ES-03 platform, the issue when implementing a business case, is understanding the workload or workloads involved and running them on the most suitable platform, considering the holistic needs of all stakeholders. This is primarily a business discussion – if the business case is for modernising a legacy application, the important thing is not to go out of business on the migration journey. In some cases, this might mean that a “good enough” destination is preferable overall to an “optimal” destination. You can waste a lot of time and resources deciding what “optimal” means; a really good, testing, definition of “good enough” may be easier to achieve.

The issues that you may meet during modernisation include the identification of all stakeholders, and weighting them by their importance; and making an unbiased business case, without undue influence from individual self-interest, politics, or a religious belief in a particular technology. And, as Bloor Analyst **Martin Banks has said**, “The issue here is that focusing on technology changes misses the point that the business processes don’t get changed. If they do, they are changed because of the technology change, which is entirely the wrong way round.” Ideally, you need a reasonably mature organisation; one that makes fact-based decisions, one that adopts some form of “do – measure – evaluate – improve” approach; and one that definitely avoids the seduction of “blame culture”. Sometimes, modernisation involves moving platforms, sometimes it doesn’t, but the decision should be fact

based and take a lifetime perspective. What is a cost effective decision with a six month horizon may cripple the organisation’s productivity over the next decade.

It is important, as DXC explained to us, to balance value and cost arguments. Value is hard to quantify in advance, costs are easy to pin down in monetary terms. This often leads to the prioritisation of cost over value – the danger is that by cutting investment to the bone you start to impact productivity and leave no spare capacity for innovation and experimentation. It is hard to be a mutable business if there is no slack anywhere

Part of DXC’s value proposition is skills transfer for optimising an ES-3 platform’s service costs and productivity, using the management expertise, processes and tools it has built up over the years. For instance, a public sector business needed a more agile operational platform than it had on-premises. Platform-as-a-service delivered improved performance (transaction volumes improved ten times and response times improved by 97%) and reduced the risk of outages and service disruptions resulting from capacity increases.

Another use case is from an automotive manufacturer with a business need for an infrastructure outsourcing contract and lower costs. It saved up to 30% on total cost of ownership with a well-chosen multi-year infrastructure outsourcing contract offering less risk than other competitors. Fast provisioning by itself resulted in savings of \$1.3 million.

An insurance business needed to scale the business from a regional to national provider; with 3rd party help, it achieved a robust, flexible technology environment that supported the company’s growth goals with stable applications, ES-3 operations, and flexible resources, whilst still reducing the cost of IT by 50% in five years. The goal is for organisations to accommodate organisational and market changes without incurring significant capital investment.

Nevertheless, there are real issues to address with many legacy ES-3 installations, which you wouldn't expect to encounter if you had the luxury of a net-new implementation. In part, these are because of the importance of these legacy systems to the business. They are often regulated "systems of record" and often have to be kept for many years. If someone queries a transaction from five years ago, can you recreate it, if you are now running on a different platform? Of course, you can and should include answering such questions in your design, but satisfying this requirement may add to the cost of migration.

A related question is that of maintaining the behaviours of your systems on a new platform. If you are reporting to an external regulator, they may take exception to seeing a step change in your reporting as you switch to a new platform – even if the new figures are, in fact, more accurate. Will you have to account for many years of supplying the "wrong" figures? Even a small error could add up to a large sum in a big system. What this means is that when you engage with external consultants for migrating off ES-3, the cost savings look impressive. But once you are committed, you get told about "regression testing", to prove the outcomes have only changed in ways that you attended – and the cost saving can markedly reduce.

Issues that you may have to consider include:

- Pressure from the rising cost of older, less mainstream, infrastructure.
- Older, often monolithic designs may add operational complexity that impacts the operation of the business.
- Fewer skilled personnel who understand older systems – issues of death, retirement and career development
- A consequent need to reduce dependencies on systems that do not offer modern environments (so you might want to move from a traditional Mainframe to zLinux);
- Workload optimisation: ES-3 works best when highly utilised and often offers cheap (or free) processing power on "speciality engines" for Linux, cryptography etc., which you need to take advantage of. This means that you may want to move workloads onto ES-3 if it is available.
- The rise of low-cost hosting solutions and "as a service" delivery as an alternative on-premises infrastructure mitigates the increasing challenge of managing mainframes in-house.
- Cultural issues including resistance to unfamiliar technology but also addressing "blame cultures" and "hero cultures" that make implementing fact-based changes harder. We deal with these in more detail below.
- The difficulty of finding specialist, skilled, mainframe personnel these days.
- Knowledge management: the mining of the business knowledge embodied in many legacy systems for re-use and the capture of knowledge/skills in the heads of retiring ES-3 specialists. You may need to outsource knowledge and advisory services to a trusted third party if they are no longer available in-house.

In general, you address these issues by designing a modernisation process with well-specified success indicators and review points, probably with the help of experienced external advisors. It is important that you budget (money and resources) for change management, cultural development, training and knowledge management – and don't overlook investing in modern tools such as machine learning. Don't expect cultural change, say, to "just happen"; and don't expect people to retrain in their spare time.

Application Modernisation – technology issues

The business case for modernisation should come first. There is little point in modernising an application which



If someone queries a transaction from five years ago, can you recreate it, if you are now running on a different platform? Of course, you can and should include answering such questions in your design, but satisfying this requirement may add to the cost of migration.





Even Linux, while fundamentally a far better design than Windows, was never really designed in the first place for the sort of workloads z/OS excels at.



is working “*well enough*” where it is, beyond (possibly) adding APIs to extend its scope. Nevertheless, when “*businesses run on software*” it isn’t possible to ignore enabling technology and many existing ES-3 applications may be both business critical and, being written in old-fashioned monolithic (but, hopefully, component-based) form, less part of the mainstream IT of the organisation than they should be – which makes them prime candidates for modernisation.

Before you start, however, think about your investment in ES-3. This platform is so powerful, scalable and resilient, not just because of the physical hardware you can see, which is impressive, but not uniquely so. To a significant extent, the value of legacy ES-3 platforms comes from the sophistication and design of their operating systems and their integration with chip architectures. IBM’s z/OS operating system on its Z ES-3 platforms is a typical implementation of ES-3 using proprietary IBM chips, but the 8086 (PC) architecture capability has improved over the last 20 years and has enabled Unisys, particularly, to provide emulation of the expected **ES-3 capabilities** on this platform too.

Don’t overlook the importance of the operating system. IBM’s z/OS has its basis in the old OS/370 architecture (and before that OS/360) and has lasted because its original designers got it right. It was designed for performance, security, user isolation etc. from the start. In contrast, early Windows compromised its hardware abstraction layer with device drivers early on. What this means is, to speed up access to printers, disks etc., they were allowed direct access to the hardware and the Windows operating system kernel. Although things are a lot better now, Windows has been playing catch-up [[DOWNLOAD PDF](#)] with bolted-on security and fundamental redesigns ever since. Think of Windows Vista and Windows 10 and the upheaval they caused while addressing the security failings of Windows 95. Even

Linux, while fundamentally a far better design than Windows, was never really designed in the first place for the sort of workloads z/OS excels at (paradoxically, if you do want to run Linux, ES-3 may be the best place to do it) and **still has some problems**.

Another technical benefit of ES-3 is the design of much of its software. There is a reason that businesses are still running on IBM’s CICS (**Customer Information Control System**) Transaction Server after some 53 years – and think of the aggregate business value it has delivered over those decades. Parts of CICS were fundamentally designed properly (in a very real sense “*correctly*”) in the first place, using mathematical proof for key components [[DOWNLOAD PDF](#)], which is something that developers of fundamental software components that will be widely reused could **learn from today**. The trick, we think, is to combine **mathematical proof**, used where it works well, with conventional testing, used where it doesn’t.

There is also a connected issue: even with all the current furore in the technical press about microservices-architecture, this may not always be the best design for applications. A microservices-architecture is where applications are built up from small, independent, easy-to-write modules, orchestrated with lightweight network calls to deliver business outcomes. This has real benefits for coding efficiency and quality, and fits well with modern developments such as containerisation. Nevertheless, microservices also have shortcomings such as involving many more moving parts, leading to difficulty in designing and testing the system as a whole and an increase in infrastructure complexity.

We wouldn’t go so far as to recommend absolutely monolithic applications (written as just one slab of unstructured code; they can perform very well but are often the devil to maintain and enhance), as we spent a lot of the 1970s and 80s promoting component-based architectures and

reusable components. Component-based design works rather well in COBOL, the Common Business-Oriented Language used for programming back then for what we'd now think of as monolithic applications. Nevertheless, microservices-based architectures are generally the first choice for today's systems as they are easier to code and test as individual services; even if they are also harder to integration-test and more complex as a whole system. Each approach has its place, and both can be disastrous if used with a poor architecture or if designed badly. It is, as usual, a question of matching the available approaches to the environment and business need. If you want a more technical look at the issues here, which is out of scope for this paper, there are many papers dealing with the pros and cons of micro-service architectures on the Web.

Modernisation may, but need not, involve moving some functionality off an ES-3 platform. There are now key (and well tested) technologies supporting application modernisation from companies such as Micro Focus, Rocket, DXC, IBM and so on. More generally, new enabling technologies are now available:

- **Hybrid multi-cloud.** This allows you to treat all platforms as one, using cloud-style APIs for access. With hybrid multicloud, virtualised infrastructure can reside on public cloud, private cloud, on-premises private cloud. You can even treat systems running on-premises ES-3 hardware as a corner of your cloud with a particularly restrictive SLA, as long as it is accessed through cloud-like APIs.
- **Containerisation** (with, for example, **OpenShift** and **Kubernetes**). Without going into technical details, containers are light-weight virtual machines, which can encapsulate existing components; and kubernetes is an open source industry standard for the orchestration of containers into business applications.
- **The Open Mainframe Project**, is the home of the Zowe open source software framework that should facilitate the management and control of software on the Z platform with a user experience familiar to Cloud users generally. We find Zowe very interesting in concept, but we haven't seen it used significantly in production yet.
- **ES-3 DevOps** As the barriers between Dev and Operations fall and with solutions from vendors like BMC and initiatives such as the Open Mainframe Project, ES-3 development and operations teams can increasingly manage, control and develop software on, say, the Z platform just as they would on any other cloud platform.
- **LinuxONE**, a IBM ES-3 platform specialised for Linux, which scales up to millions of sessions, tens of thousands of containers, 8,000 virtual servers and over 30 billion RESTful web interactions per day. Linux also runs on any of IBM's z/OS platforms.
- **AI** – IBM Research is using AI (augmented intelligence) to automate refactoring (modernisation) of monolithic Java applications into modular components. This is not available for other languages yet, but it is a pointer to the way IBM, and others, are thinking about refactoring monolithic applications on z/OS.

One key technology issue is testing and, especially, as already mentioned, regression testing. In theory, every requirement has a test or several tests – if a requirement can't be tested, it isn't a real requirement. I can't ask for a system that performs fast enough, as that is a matter of opinion. If I ask for it to maintain sub-second response times on 90% of transactions, with no transaction longer than 2 secs, while processing up to 2000 transactions a minute, that can be tested.



Microservices-based architectures are generally the first choice for today's systems as they are easier to code and test as individual services; although they are also harder to integration-test and more complex as a whole system.





The bottom line is, don't assume that you understand ES-3 technology issues and options today because you used a green-screen mainframe many years ago. The ES-3 technology has evolved over the years just as the rest of IT has.



Unfortunately, if you are modernising an older system, you may not have its original test cases and test data; if you have, they may no longer run because of changes to the system. You will have to baseline the existing system (document its behaviour and performance), design and implement suitable test cases (while making sure that the privacy of any real customers that might be included in your test data is protected) and build a test pack. This is the equivalent of coding quite a large system and the resources needed may render replacing legacy systems from scratch impracticable. Modernisation may be a better choice than wholesale replacement simply because the less you change, the less you have to test. Regression testing, to prove that the behaviours of a legacy system haven't changed except as anticipated, is a problem for modernisation, especially if the original analysis, design and requirements documentation is no longer available (there are working systems where even the source code doesn't correspond exactly to what is being executed). It is an even bigger problem if you rewrite from scratch, because you not only have to find out what the requirements today are, you have to determine how the old system behaved in equivalent detail, so you can determine if there are any changes, and address them, before anyone outside the organisation notices.

Regression testing is really part of managing change risk. However good and well-engineered a new platform is, there is always the risk that something has been left out of its design spec. or it behaves slightly differently to how you expect it to. It's a balance – a new, more modern, platform may well be more cost-effective than a legacy platform – but there is a risk-related cost of migration too. When you are evaluating the cost of new technology and new hardware, don't forget to include migration cost and risk management, unless you are building an entirely net-new system.

In summary, there are many more ES-3 platform options than there used to be.

You can access Mainframe-as-a-Service (MaaS) using an ES-3 platform running in someone else's datacentre, via cloud services, if that makes more sense than running your own datacentre. Another example: you can run Linux on z/OS, but you can also run it natively on a system x bladeserver plugged into the back of your ES-3 hardware.

Even pricing has got more flexible in many cases – no longer do you always have to install (and pay for) a bigger ES-3 platform than you need, in case you have to deal with, say, a pre-Xmas peak in demand. You can often install extra processors or a processor speed switch which lets you pay for what you are actually using and switch on extra power just for when you need it – and then switch it off and stop paying for it. Another innovation these days is flexible mainframe pricing. This gives ES-3 customers more flexible pricing models (by average CPU usage or machine capacity) than are traditionally available. See for example, [*Precisely's blog*](#).

In addition, a particular capability on many ES-3 platforms is the availability of "speciality engines": hardware assisted services on specialist processors for things like running Linux, computation-heavy processes such as encryption and so on. ES-3 handles multiprocessing particularly well and these processors are sometimes effectively free or involve a one-time cost and are critical to getting the best possible value out of your mainframe platform – as [*Precisely explains*](#).

The bottom line is, don't assume that you understand ES-3 technology issues and options today because you used a green-screen mainframe many years ago. The ES-3 technology has evolved over the years just as the rest of IT has.

Enterprise Server 3.0 and DevOps

DevOps is about driving the business value of software delivery, according to [*"The IT Manager's Guide to DevOps from digital.ai"*](#). DevOps has now mostly taken over the development mindset in distributed

systems, although iterative variations on the **Waterfall method** are still used effectively in some outsourcing-style engagements. DevOps is, in essence, the removal of silo boundaries between Development and Operations, with a continuous delivery pipeline for delivering high-quality software to production, quality being assured by continuous integration and continuous testing. There is, in essence, a positive feedback loop between software build and production release.

This is not the place to examine the issues and success factors for DevOps in detail. For instance, whether Dev or Ops predominates. We would say Ops over Dev, since Ops looks after the lifetime delivery of value from software, which needn't be built (making Dev less essential) but could be purchased as containerised components. Then, there is the issue of whether the feedback loop is big enough, since what business needs is not just high-quality software but the right software – perhaps the pipeline should start with design thinking and deliver business outcome or business user experience. And, should non-functional requirements such as security and performance be built into the pipeline – of course they should.

This is relevant to ES-3 because DevOps is proving a key way of bringing the ES-3 culture, and particularly ES-3 Ops culture, into the organisation's mainstream. This is why DevOps has been given a section on its own here. It is also why the expanded view of DevOps above matters, because ES-3 culture is based around very large systems, optimised throughput and business outcomes, not just software delivery. That said, DevOps addresses some endemic issues with ES-3 software delivery in the past – lack of change velocity; lack of flexibility; and a mismatch between highly regulated ES-3 Systems of Record and the modern Apps which power user experience (both can be maintained with a DevOps culture). ES-3 DevOps is now becoming ubiquitous and the default approach to building and

maintaining ES-3 systems (although, as I said, there can still be a place for iterative waterfall). Compuware can claim a lot of the credit for this, and it is a good place **to find out more about it**. ES-3 DevOps remains a strong part of BMC's vision after its acquisition of Compuware, although BMC and Compuware no longer have a monopoly of ES-3 DevOps.

Why ES-3 DevOps? Well, you should aim to achieve 5 positive outcomes from its adoption:

- 1 More Mutable ES-3 systems – systems that can support continuous evolution in support of changing business needs;
- 2 A greater velocity of software delivery – small functional changes delivered often;
- 3 Better quality software from continuous delivery/testing;
- 4 Better business outcomes from better quality software;
- 5 Better integration between teams working on Systems of Record and on user interface Apps.

If you are not achieving these, and this implies that suitable metrics are available for determining this, then you should be working out why not.

Enterprise Server 3.0 and Systems Management

Systems management is both the USP for ES-3 and a perceived barrier to adoption. ES3 is supremely resilient, robust, secure, performant and manageable – providing it is, in fact, managed. But there are real differences with ES-3 from distributed systems: ES-3 tends to perform better when very highly utilised – **one set of benchmarks** found average mainframe CPU utilization over a 30-day period was 53%, with the best-practice participant averaging 75% utilisation. During prime shift (Monday through Friday, 7:00 a.m. to 6:00 p.m.), survey participants achieved average utilisation of 68%, with the best-practice participant reaching a high of 83%. This contrasts with PC servers,

“
ES-3 DevOps is now becoming ubiquitous and the default approach to building and maintaining ES-3 systems.
”



Adequate ES-3 systems management training must be available, and you might even want to offer retiring ES-3 support staff a part-time role as mentors for new staff. Nevertheless, a lot of distributed management knowledge translates over to ES-3 well; perhaps one of the biggest, yet least important, barriers to acceptance of ES-3 is terminology.



which usually work best with much lower utilisation. And, you can't easily just plug in a new ES-3 when you run out of capacity, in the way in which you can plug a new commodity Intel server into your cluster. But, these days, IBM (for example) will sell you (or give you access to) a much larger Z ES-3 server than you currently need and only charge you for the functionality you actually use. In fact, IBM will switch on (and charge for) extra power to cope with a processing peak that only lasts for a few days, after which charging reverts to normal.

What this means is that adequate ES-3 systems management training must be available, and you might even want to offer retiring ES-3 support staff a part-time role as mentors for new staff. Nevertheless, a lot of distributed management knowledge translates over to ES-3 well; perhaps one of the biggest, yet least important, barriers to acceptance of ES-3 is terminology: the use of terms such as DASD (Direct Access Storage Device – or disk drive), Workload, JCL (job control language) etc. This is easily addressed with training and online support – but it does need to be addressed.

Some of the management skills you may need for an effective ES-3 installation (beyond the general system management skills you probably already have) include:

- **Job scheduling.** This is EC-3's "secret sauce" – job schedulers aren't widely used in distributed systems (beyond ad-hoc scripting) but they are a mature technology on ES-3. If you want to maximise throughput of very large workloads on ES-3, you want effective parallel processing. Running a network of independent jobs in such a way that there is no competition for resources is an effective way of maximising parallel processing and a good ES-3 scheduler is how you manage this.
- **Capacity planning.** If you are processing very large workloads you probably can't afford to install a great deal of spare capacity "just in case", so capacity planning is essential.

- **Workload management.** If you need to put workloads on the most suitable, cost effective, platform, you had better understand workloads and how their characteristics change over time. Workloads may well need to migrate to different platforms as their characteristics change.

- **Performance management.** ES-3 can be very performant but they are as susceptible to bottlenecks and so on as anything else. Since you are probably using ES-3 because you need the performance it is capable of, effective performance management is a good idea.

- **Security.** ES-3 can be made extremely secure indeed, but some people tend to rely on "security by obscurity", because "no-one knows how ES-3 works". Believe me, if your ES-3 platform holds valuable information they want to steal, criminals can hire ES-3 experts. Moreover, there are published ES-3 exploits which require no specialised ES-3 knowledge. You need to ensure that your general security policies and best practices are extended to cover ES-3.

- **Pricing models** – and, in particular, recent innovations in pricing

The importance of keeping up with the latest ES-3 trends cannot be overstressed. Modern mainframes are smaller and tell a good "green" story: water cooling, for example, facilitates the reclamation of waste heat and IBM was using this technology in a Zurich supercomputer as far back as 2010. And there is now even an open source Open Mainframe initiative. It held a virtual Summit in 2020, with leadership from the Linux Foundation event team; and sponsorship from Broadcom, IBM, Rocket Software, SUSE, Vicom Infinity, and Micro Focus; and more than 380 registrants from 176 companies. The open Mainframe project is intended to promote the use of Linux operating systems and

open source software on ES-3 platforms. Its first project is Zowe which aims at making IBM's z/OS more accessible with a choice of Command Line Interface; or the "Zowe Explorer" Visual Studio extension (a web browser from the Zowe Application Framework); or through REST APIs and web sockets served through the API Mediation Layer. Zowe is also an easy-to-extend tool plugin platform, although existing ES-3 DevOps solutions (such as those from BMC) can already be used to connect the ES-3 platform with popular Open Source tools, such as Git and Jenkins.

The bottom line is that ES-3 is by no means a "dying platform". Interesting systems management developments are a good sign of there being life in the "old" platform yet – and a modern ES-3 box even looks sexy!

“

...a modern
ES-3 box even
looks sexy!

”



Cultural issues around Enterprise Server 3.0



The biggest issues you'll have with modernising ES-3 platforms and making them just another part of your general IT infrastructure are probably cultural.



The biggest issues you'll have with modernising ES-3 platforms and making them just another part of your general IT infrastructure are probably cultural. The traditional ES-3 culture is risk-averse and outcome-focused, fully aware that something going wrong that impacts the business can lose it lots of money; distributed culture is innovation oriented and accepts risk, even at a cost to the business. You need both and part of what management does is balancing the safety of the status quo against innovation and risk – depending on the organisation's appetite for risk.

Prime directions for cultural modernisation are the removal of silos and the consequent barriers to communication and collaboration; and consideration of the needs of all stakeholders (including society – green computing – and auditors, customers and regulators), not just the IT group. Some of these groups are more risk-averse than others, some are more fickle than others, some more financially driven than others. And silos may exist because of the politics of strong personalities or because the organisational culture is built around *“command and control”*. Cultural modernisation needs thought and skill – it isn't something trivial that can be managed in someone's spare time.

Cultural change must be specifically part of the organisation's plans, with allocated resources, and it should appear in the budget. If it doesn't, we'd question whether there is real management commitment to the importance of culture and doubt if anything will change – or, at least, whether any changes will be *“sticky”* after top management attention moves on. Cultural change does need top management commitment but it can't simply be imposed by diktat. Bottom up buy-in to change is important and it is often necessary to get independent external advice and provide mentors with experience of both where you are coming from and where you are going

to on the change journey. In house change managers often don't have the depth of experience needed and can be hampered by a fixation on the status quo and internal politics. Don't overlook the wealth of knowledge available from vendors, but it is probably best not to rely on vendor support – sometimes it may be biased towards their own products and agendas; and even if it isn't, your staff and other stakeholders may not believe this.

If you do use external change mentors, don't overlook the need for skills transfer. You don't want to be locked into external consultancy for ever. But you also want to see cultural change as a continuing programme, not as a project with an end date, and you want to back it up with *“cultural awareness”* training that starts with the induction of new staff. Culture isn't simply a matter of retraining ES-3 staff either, it is a matter for all staff and you must explain what the benefit is for them – for instance, a focus on diversity and resilience will probably improve the user experience for everyone. Career development is a part of managing culture, as is diversity and social inclusion.

Good culture attracts good talent, so after you have modernised your culture, put diversity commitments, privacy policies and commitments to social responsibility on your website. Don't ever try to fake them (this is very counterproductive, even to internal morale) but if they are genuine, they help you to be seen as a good place to work. Having a VP of Diversity or Social Responsibility, if genuine, might help too. Good culture isn't just a nice thing to have, it brings real business benefits for quality and collaboration as well.

Customer use case

1

Integration of new digital services with core ES-3 systems for a very large US insurance company.

Risk Factors (in priority order)

- 1 Cultural change, integration of Agile and previously non-Agile communities.
- 2 Communication across diverse physical infrastructure platforms.

Actors

Architecture managers.

Priority

Strategic to future success of the organisation.

Status

In production implementation.

Challenge

This organisation wants to accelerate velocity of systems innovation and introduce consistent and modern development practices across all of its development environments, including ES-3 (IBM Z).

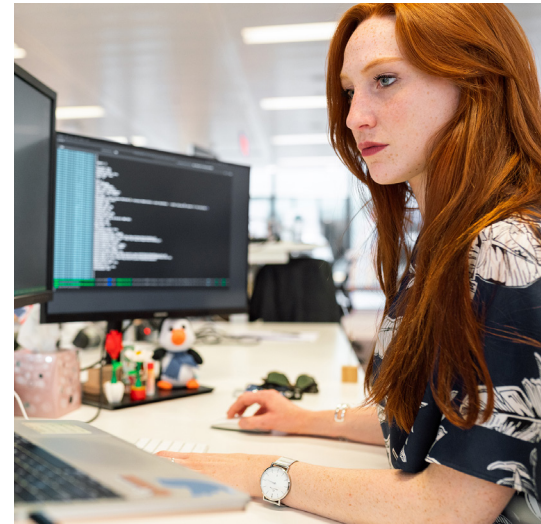
Its customers now expect a consistent user experience and ease of goal achievement regardless of whether they engage face-to-face, online or via a mobile app. It is moving to offering self-service tools, so time-to-market matters and development practices must support rapid change.

The organisation sees its core ES-3 systems, tried and tested over 50 years, as offering competitive advantage, but also wants to compete on being nimble and responsive. It is achieving this, in part, by introducing DevOps practices to its ES-3 platforms, thus enabling all its developers to share a common culture.

Consequences

This organisation eliminated barriers to change on the IBM Z platform, with one of the first changes being to replace a legacy Source Code Management system with Git. It sees modern development tools as helping to open up the ES-03 platform to younger developers with no existing experience of IBM Z.

It is building on its successful high-availability ES-3 platform, which is essential to its business and will continue on into the future, with a new hybrid-cloud infrastructure, which supports a platform agnostic future for business automation – which each workload running on the most appropriate platform for delivering the desired business outcome. And, DevOps is allowing it to accelerate the software development lifecycle, regardless of platform.



Customer use case

2

A US State Court needs the 24x7 security and resilience of ES-3, but modernises to reduce cost of ownership.



Photo © BMC

Risk Factors (in priority order)

- 1 Change is always risky, especially when politics might be involved.
- 2 Costs in government are always in the public eye and mistakes are very visible.

Actors

IT Director.

Priority

Mission critical – a 4x7 resilient high availability computing service is essential to maintaining court services.

Status

In production implementation.

Challenge

This institution had an effective working ES-3 platform based around two ES-3 servers. It wanted to consolidate, for improved cost-efficiency and manageability, without impacting the service being provided.

The second machine was acquired for an externally-funded system and wasn't necessary for the organisations assessed business continuity requirements.

The organisation, with the help of external consultancy, determined that moving to the latest generation of IBM Z server would both boost performance and reduce footprint – the ES-3 platform is not standing still – while reducing total cost of ownership (TCO). It is worth noting that part of its success comes from optimising the new ES-3 environment. Just copying applications from one platform to another would not be so effective, so that external consultancy is often worth the investment.

Using Java and IBM Websphere, it can now provide a modern user-friendly GUI experience for its existing applications, without moving them to Linux, although it does have Linux applications on its ES-3 platform. It also has Linux applications on distributed systems, consuming data from the ES-3 core. It's new, simplified, platform offers the possibility of further application simplification in future.

Consequences

The new ES-3 platform effectively cuts maintenance costs in half and, as existing workloads can be accommodated with less CPU resource on a more modern machine, software licensing costs have reduced too.

Performance has been maintained, despite lower costs – in fact, this organisation cut overnight batch processing times (it is critical that these jobs finish before the online day starts) by about 1.5 hours, even before tuning.

It also sees the new platform as enabling valuable business initiatives such as pervasive encryption of sensitive data. And, the availability of Linux on Z makes it easier for the organisation to attract new talent (it also supports ES-3 education at universities across the USA)

Customer use case

3

Major Chinese air travel and tourism provider renews commitment to ES-3 after COVID crisis.

Risk Factors (in priority order)

- 1 Increased load following COVID 19 recovery.
- 2 “Fashion” pressure to modernise platforms radically post-crisis.

Actors

Senior management team.

Priority

Essential to taking advantage of COVID 19 recovery.

Status

Future commitment.

Challenge

We’ve always seen China as less sceptical of ES-3 than some countries, probably because Chinese culture is more inclined to accept corporate hegemony and less impressed by individual “pirates”; and also because of the sheer scale of internal Chinese systems.

In 2020, the Civil Aviation Administration of China (CAAC) reported 13.26 million air trips during the eight-day National Day and Mid-Autumn Festival holiday October 1-8, 2020, representing a 91.07% recovery compared to the same period in 2019.

In response to expected recovery loads, this firm chose to enhance its Unisys Clearpath Forward systems, as opposed to moving to a different platform, representing real confidence in the Unisys ES-3 platform.

Consequences

According to an SVP, at this company, it maintains and operates several critical passenger and cargo solutions used by China’s air travel and tourism industries including the Global Distribution System, Reservation System, Departure and Cargo systems. As China’s aviation industry continues to rapidly recover after COVID-19, the additional processing capacity within the scalable, secure and flexible ClearPath Forward environment will enable it to process more transactions – such as reservations and load calculations critical to flight safety – quickly and securely.

Key to the choice of the Unisys ClearPath Forward operating environment is its unmatched security record – it claims to be the only ES-3 platform on the market from which data has never been forcibly extracted.



Vendor Survey

This annotated list of overview vendor solutions, with a sentence or two on each, describes the vendor approach to modernising legacy from (where possible) a Business (rather than a technology) point of view, and is largely based on the [IBM Z 2021 Solutions Directory](#).

There is also a strong [IBM Z and LinuxONE Community](#), if you want to learn more about IBM solutions.

Note that these classifications are not absolute, most vendors play in many or all categories, but we have attempted to categorise by what we see as a main focus.

Passively integrating ES-3 legacy into modern systems; i.e. just extracting its data for use elsewhere.

You need to worry about data duplication – one source of the truth – and data latency – how up to date it is. This is really out of scope for this report (it is covered elsewhere in Bloor) but mentioned just for completeness.

ES-3 Education and consultancy

- [Enterprise Performance Strategies Inc. \(EPS\)](#) facilitates and educates around performance management.
- [IBM TechU](#) offers a variety of ways to learn, grow and connect with ES-3.
- [Interskill Learning](#) offers ES-3 workforce training online.
- [Trident Services Inc.](#) offers z/OS consultancy.

ES-3 Systems Management and Outsourcing

- [ASG](#) (now becoming part of Rocket) offers a range of ES-3 management solutions including cross-platform job scheduling with ASG-ZEKE, and performance monitoring/optimisation with ASG-TMON.
- [Beta Systems](#) provides ES-3 system log analysis and archive tools as well as bulk document management and archiving, automated process management and so on.

- [BMC](#) is one of the ES-3 market leaders. It has been actively extending its offerings across both ES-3 and distributed systems in the 21st century. Perhaps its most interesting focus is on leveraging augmented intelligence (AI), machine learning and predictive analytics, in order to ensure performance, availability, and optimisation across the evolving ES-3 environment.
- [DXC.technology](#) is an important ES-3 services and management company with grew out of the merger of CSC and the Enterprise Services business of Hewlett Packard Enterprise. Often collaborates with Micro Focus.
- [International Software Company \(ISC\)](#), for batch workload automation (E-Gen) and scheduler migration (uWME).
- [Rocket Software](#) is a privately-owned company that has acquired a number of ES-3 systems management tools, some available as Open Source. It is also acquiring ASG.
- [Unisys](#) offers assistance with cost balancing, operations admin, assistance with custom applications and management of Unisys cross-platform products, for its ClearPath Forward ES-3 platform. It supports hosted or cloud, remote management or on-premises models.
- [Virtualz](#) optimises/consolidates ES-3 software licences for significantly lower costs.
- [ZETALY](#) leverages AIOps, FinOps, Automation and Predictive Resource Planning for automation of ES-3 resource and capacity planning/optimisation.
- [Zowe](#) is an Open Systems Software project from the Linux Foundation Open Mainframe initiative, which (along with Red Hat Ansible automation) is bringing a cloud-like experience to Z.

Modernising legacy systems with DevOps, integrating them with other systems via APIs and building new (e.g. COBOL) code, new use cases etc.

- **ASG** (now becoming part of Rocket) claims to revolutionise ES-3 DevOps with a toolchain that operates from ES-3 to cloud and is accessed from a modern browser-based interface.
- **BMC** is a market-leading ES-3 vendor providing solutions that enable developers and operations personnel to make applications and operations on the ES-3 platform as agile and as modern as any other cloud and distributed platform is. BMC AMI and BMC Compuware provide a DevOps toolchain for the ES-3 platform that supports an open ecosystem covering the whole IT lifecycle. This enables the enterprise to operate in a consistent, automated and efficient manner across all platforms.
- **Broadcom** is a market-leading ES-3 vendor which offers a lot of application development tools under its CA brand, such as CA Brightside (an OSS application development solution using the Zowe framework and extending supports to the Code4z mainframe extension pack for the widely-used Visual Studio Code and Eclipse Che IDEs. It also provides DevOps integrations for the CA Endevor configuration management solution.
- **IBM** is a strong supporter for Design Thinking, Agile Devops, augmented intelligence and development generally (including OpenSource solutions) on the ES-3 platform. Following IBM's pivot to Open Source with its Red Hat acquisition, it has a strong integration story around Cloud Paks, OpenShift, zLinux and Linux ONE.
- **HostBridge Technology** HostBridge makes legacy CICS applications available as web services, using open standards such as JavaScript, XML, JSON, or HTTPS, without changing mainframe code.
- **Infotel** supplies a DevOps solution for quickly spinning up testing container environments on IBM® z/OS – Application Parallel Testing (APT)
- **Micro Focus** is an important ES-3 tools, services and management company with an extremely wide range of tools including important COBOL programming tools and environments as well as some Hewlett Packard acquisitions. Micro Focus often collaborates with DXC.
- **PKS** produces eXplain for IBM Mainframe for code analysis and quality improvement, especially where monolithic code has poor documentation (it supports COBOL, PL/I, Assembler, Natural, Adabas, CICS, IMS DC, IMS DB, Db2, JCL).
- **Precisely** is the new company formed through Syncsort's acquisition of the Pitney Bowes Software & Data business. It offers a range of solutions for integrating today's infrastructure with tomorrow's technology (including some data management and data quality tools which are outside the scope of this paper).
- **Rocket Software** is an important but often overlooked privately-owned company with a strong IBM provenance and a large range of ES-3 tools (including lifecycle management and DevOps) for modernising the Z platform. It supports deep analytics and augmented intelligence. It is also acquiring ASG.
- **Software AG** sells the WebMethods Platform, which aims to include all the API management capabilities necessary for the end-to-end lifecycle management of high-quality APIs.
- **UNICOM Global** modernises ES-3 with consumer-grade user experience using its UniGWdigital transformation platform to seamlessly connect ES-3 applications with web, mobile, chat and AI technologies.
- **Unisys** offers its clients help with preserving existing application investments and with prioritising and delivering modernisation initiatives and cost reduction. There is also a ClearPath Extension Kit, supporting modern languages and tools, such as Python and Git.

ES-3 Hardware and Net-new ES-3 systems

- **Amazon Graviton 2.** This is a specialised Amazon Web Services server using 64-bit Arm Neoverse cores to deliver the best price performance for Amazon EC2 cloud workloads.
- **IBM Z.** This is the archetypal ES-3 server from IBM and is a focus for ES-3 innovation and hybrid multicloud integration, influenced by IBM's pivot to Open Source and its Red Hat acquisition. IBM's ES-3 (Mainframe) revenue grew well over 50% year-on-year in 4th qtr 2020 its cloud revenue growth, across all segments, was healthy too [[DOWNLOAD PDF](#)].
- **Unisys ClearPath.** This is a classic ES-3 server running one of two operating systems: MCP, which came from Burroughs, and OS 2200, which came from Sperry; and now implemented on x-86 hardware and Public Cloud showing that IBM Z isn't the only ES-3 game in town.

Replacing legacy systems (eliminating legacy servers completely)

- **Astadia** says on its website tagline, "We're consultants on a mission to free organizations from their mainframes". It is important to run workloads in the most appropriate platform and there are certainly some workloads that are not best suited to the ES-3 platform (although perhaps rather fewer of these with Z Linux available, we'd think, these days). Astadia has successful customer stories for ES-3 replacement, for both Z and Unisys platforms, and it has the technical expertise needed for this in-house (we have talked to its CTO). We are still a bit concerned that its marketing message on its website doesn't mention regression testing much and rather implies that migration is the only solution available and that it will be easier than it may turn out to be in practice, if the workload being migrated really needs ES-3 levels of performance/

resilience. As we always say, there must be a comprehensive whole-lifetime business case for either migration or modernisation/retention, and we note that Astadia is a Micro Focus partner. We also note that Astadia has acquired (in March 2021) an automated refactoring tool, for translating systems written in programming languages like COBOL (the latest version of which is COBOL 2014), to newer programming languages such as Java and C#. We would see this as a significant capability.

- **Amazon.** There are many ways to get mainframe data onto AWS for processing – but this is often just extraction/replication, not integration. It looks to us like migration off of the Enterprise Server. For instance: How to Unleash Mainframe Data with AWS and Qlik Replicate; How to Enable Mainframe Data Analytics on AWS Using Model9; Real-Time Mainframe Data Replication to AWS with tcVISION from Treehouse Software. As soon as the Enterprise Server data is unlocked and available within an AWS data store, such as Amazon Simple Storage Service (Amazon S3), customers can use the wide array of analytics and machine learning services available on the platform for easy access to all relevant data, without, allegedly, compromising security or governance. Customers select AWS data services from data catalogue and data processing to interactive analytics, real-time analytics, operational analytics, dashboards, and data warehousing. Even innovation is possible, as once mainframe data is on AWS, customers innovate by creating new functions with cloud speed. For example, some choose to create micro-services, with a complete serverless stack via AWS Lambda, accessing their mainframe data. Others decide to make mainframe data available to new channels, such as mobile users via Amazon API Gateway or voice devices such as Amazon Alexa.

Mainframe data can also be easily moved into machine learning models. This all sounds good, but it overlooks a fundamental problem, which is probably addressable but doing so may involve considerable resources. Your AWS Data store is duplicate data. If the “one version of truth” is on the mainframe, how out-of-date is the AWS Data Store? Is latency an issue? Can people update the AWS Data Store without updating the mainframe data?

- **Fujitsu Progression.** This claims to be the answer to moving on from ES-3 and converting legacy COBOL to newer languages.
- **Google.** Google Cloud *acquired Cornerstone Technology* to assist with migrating mainframe onto Google Cloud. This reads more like rip and replace than modernisation. Cornerstone offers tools to help with planning a migration roadmap, including identifying microservices, converting code languages and databases; and automated data migration. If your workloads suit Google (not all will), this will be very useful.
- **Microsoft** has a rich toolset for migrating ES-3 to Azure. The Azure platform seems to promise something like **99.9% availability**, or about 8.75 hours downtime per year, which (although the devil will be in the details) falls far short of IBM Z platform targets. Enterprise-scale computing on Z is all about trust – *“given our global operations, we need our systems to be available round-the-clock, so reliability is a priority for us. With IBM Z, we’ve never experienced any performance or reliability problems to shake our trust in the technology”* – **Ian Wilson, Managing Director, Fort Vale**, and we’re not sure that Azure has achieved that level of trust yet.

Conclusion



...if a legacy system is still fit for purpose, providing a service-based web API may be sufficient modernisation.



It is clear that there is still considerable reliance on ES-3 systems and data in the very largest enterprise operations and that moving off these systems is non-trivial – partly because of their tried and tested capabilities when scale, resilience and throughput are critical, and partly because proving that behaviours haven't changed, except in anticipated ways, on a new platform needs a lot of regression testing. If you want some, possibly surprising, statistics around the importance of E3-3 (mainframes) in business today, ***Precisely collected some***, with information sources, around June 2020. Organisations are increasingly moving away from migrations based solely on fashion and platform dogma towards modernisation, founded on a documented business case. This is made easier by the move towards hybrid multicloud, containerisation and APIs – if a legacy system is still fit for purpose, providing a service-based web API may be sufficient modernisation.

Less obvious issues you might have to consider include:

- Managing culture change, the ES-3 Culture is risk averse and tends to prize reliable service delivery above innovation (and especially above innovation for its own sake);
- The business and operational knowledge held in the heads of long-term ES-3 employees, losing which is wasteful;
- Supporting all stakeholders, in the business not just in IT, of course, but also including regulators (internal and external), business continuity, security, archive, audit and so on.

When you are modernising ES-3, don't forget that ES-3 is most cost-effective when heavily used. So, if it is satisfying a real business need, don't overlook the advantages of moving more workload onto it – perhaps by using it to support Linux virtual machine environments, which it does very well.

As Bloor analyst ***Martin Banks*** has put it: *"in practice the process of business transformation comes back to basic issues such as 'Why you want to do it in the first place? What are the goals?'. These make up the map that must then be followed, with analysis of the data flows through applications detailing the paths and the pitfalls. Only then is it possible to decide which applications can be retired, which re-written and what is required that is new. Only then will issues about technology come clear and make sense at a time when so many vendors still seem intent on producing solutions in search of a problem."*

Finally, your strategic aim should be to make ES-3 "just another computer" and ensure that the organisation's service and security policies apply equally to ES-3 platforms as to everywhere else. To do this, you may need new technology (such as AI and machine learning) to help bridge any skill gaps between various technical teams; and you may need to be very familiar with both old and new technology, in order to produce a functioning whole. We strongly suggest getting assistance from independent 3rd party operations as well as your existing vendors, as they can bring broader experience to the process than you may have in-house. They can also be a source of experienced mentors, to help with managing the change process.

FURTHER INFORMATION

Further information about this subject is available from www.bloorresearch.com/update/2645



About the author

DAVID NORFOLK

**Practice Leader:
Development & Governance**

David Norfolk was working in the Research School of Chemistry at the Australian National University in the 1970s, when he discovered that computers could deliver misleading answers, even when programmed by very clever people. His ongoing interest in getting computers to deliver useful automation culminated in his joining Bloor in 2007 and taking on the development brief.

Development here refers to developing automated business outcomes, not just coding. It also covers the processes behind automation and the people issues associated with implementing it. He sees organisational maturity as a prerequisite for implementing effective (measured) process automation and ITIL as a useful framework for automated service delivery. He also looks after Collaboration and Business Process Management for Bloor, and takes a lively interest in the reinvention of the Mainframe as an Enterprise Server.

David has an honours degree in Chemistry, a graduate qualification in Computing, and is a Chartered IT Professional. He has a somewhat rusty NetWare 5 CNE certification and is a Member of the British Computer Society (he is on the committee of its Configuration Management Specialist Group).

He has worked in database administration (DBA) and operations research for the Australian Public Service in Canberra. David then worked for Bank of America and Swiss Bank Corporation in the UK, holding positions in DBA, systems development method and standards, internal control, network management, technology risk and even PC support. He was instrumental in introducing a formal systems development process for the Bank of America Global Banking product in Croydon.

In 1992 he started a new career as a professional writer and analyst. He is a past co-editor/co-owner) of Application Development Advisor and was associate editor for the launch of Register Developer. He helped organise the first London CMMI Made Practical conference in 2005 and has written for most of the major computer industry publications.

He runs his own company, David Rhys Enterprises Ltd, from his home in Chippenham, where he also indulges a keen interest in photography (he holds a Royal Photographic Society ARPS distinction).

Bloor overview

Technology is enabling rapid business evolution. The opportunities are immense but if you do not adapt then you will not survive. So in the age of *Mutable* business Evolution is Essential to your success.

We'll show you the future and help you deliver it.

Bloor brings fresh technological thinking to help you navigate complex business situations, converting challenges into new opportunities for real growth, profitability and impact.

We provide actionable strategic insight through our innovative independent technology research, advisory and consulting services. We assist companies throughout their transformation journeys to stay relevant, bringing fresh thinking to complex business situations and turning challenges into new opportunities for real growth and profitability.

For over 25 years, Bloor has assisted companies to intelligently evolve: by embracing technology to adjust their strategies and achieve the best possible outcomes. At Bloor, we will help you challenge assumptions to consistently improve and succeed.

Copyright and disclaimer

This document is copyright © 2021 Bloor. No part of this publication may be reproduced by any method whatsoever without the prior consent of Bloor Research.

Due to the nature of this material, numerous hardware and software products have been mentioned by name. In the majority, if not all, of the cases, these product names are claimed as trademarks by the companies that manufacture the products. It is not Bloor Research's intent to claim these names or trademarks as our own. Likewise, company logos, graphics or screen shots have been reproduced with the consent of the owner and are subject to that owner's copyright.

Whilst every care has been taken in the preparation of this document to ensure that the information is correct, the publishers cannot accept responsibility for any errors or omissions.



