

The logo for Bloor White Paper. It features the word "Bloor" in a bold, white, sans-serif font, preceded by a stylized icon of a building or server rack. To the right of "Bloor", the words "White Paper" are written in a large, white, sans-serif font, oriented vertically.

# Bloor White Paper

White Paper by Bloor  
Author **David Norfolk**  
Publish date **September 2022**

---

## **The value of the mainframe – the modernised mainframe still has a bright future as Enterprise Server 3.0**



“

The mainframe must be modernised as “*just another – very powerful – server*”, playing an equal part in enterprise IT, and its cultural silo must be dismantled. We call the modernised mainframe Enterprise Server 3.0 (ES-3).

”

# Executive summary

**I**t is becoming recognised that the mainframe hasn't gone away and isn't going to disappear anytime soon. It is still in use (according to IBM) in almost three quarters of Fortune 500 companies; still used for almost all credit card transactions and for over two-thirds of production IT workloads – and yet it only represents 6% of IT cost. You can quibble about the figures – volumes need to be weighted by value and importance, but the fact remains that a lot of important business data for the largest companies is still stored and processed on the mainframe. Even if the end user interaction is on a mobile phone, a lot of the data involved may derive ultimately from the mainframe.

For the largest systems, it is simply easier (and probably cheaper) to provide the levels of throughput, resilience and security that a large regulated business requires on a mainframe. A Forrester report [\[here\]](#) has shown that moving the sort of workloads that need a mainframe anywhere else is seldom cost effective – in part, we think, because of the cost of proving that the behaviours of regulated systems haven't changed in unanticipated ways during the move.

There is a downside to this situation, of course. The mainframe can become a risk-averse silo, acting as a brake on the evolution of a mutable business; and there is also the issue of an aging, and retiring, skill base. The mainframe must be modernised as “just another – very powerful – server”, playing an equal part in enterprise IT, and its cultural silo must be dismantled. We call the modernised mainframe Enterprise Server 3.0 (ES-3) – 1.0 was the 1980s mainframe, 2.0 was the (incompletely delivered) 1990s clustering technology. Note that these generations don't replace each other, in the medium timescale at least, they build on what came before and (in 2022) all of these generations co-exist, although the unmodernised 1980s-style mainframe is in decline.

How does one modernise, to arrive at ES-3? Well, somewhat simplistically (we expand on this in the body of this document), there are 3 main starting points:

- Add bidirectional APIs to legacy applications, without changing the code. This is a starting point – you can take it further with containerisation and packaging legacy as micro-services, using tools such as OpenShift that run on hybrid cloud platforms on both distributed systems and mainframes, to arrive at full ES-3.
- Unlock Data Value – using ES-3 data as a source of high-quality data.
- Modernising the user experience, especially to deliver the quick wins that maintain enthusiasm for modernisation.

There is also a fourth point – managing cultural change as your mutable business evolves and satisfying the people issues being met by all of the stakeholders in your business.

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

Enterprise Server 3.0 is usually the home of mission critical and often regulated systems and business information. The custodians of these systems are usually more risk-averse than those running general IT, and with good cause (a change can adversely impact service levels and the bottom line, as well as improve things). A mutable business, on the other hand, must support change and evolution, as well as managing risk effectively. It is important, nevertheless, that ES-3 doesn't become a change-averse silo, a drag on the agility of the business.

Success implies building on a strong framework of “just enough governance”, so that mutable opportunities can be exploited with acceptable controls, but fast enough so as to meet business needs.

“

**...we hate waste,  
and we hate  
throwing away old  
investments that are  
still delivering value,  
unless there is a very  
good business case  
for doing so.**

”



Success means managing change bottom up as well as top down, if you want cultural changes and “*good practice*” to stick after management attention moves on.



Think about managing the required culture around this – instigating “guilds” providing input to development from all stakeholders (including the regulators) perhaps; rotating staff around both distributed and ES-3 environments; and, of course, formal training and mentoring (and not in employee’s own time). Cultural change must have access to resources and budget.

Success means managing change bottom up as well as top down, if you want cultural changes and “*good practice*” to stick after management attention moves on. So, this paper should be of interest to (and is written at a level appropriate for) all informed and interested stakeholders, including those in the business.

# The place of ES-3 in a mutable business

**B**loor is a long-established Industry Analyst firm supporting Mutable Businesses, in a constant state of evolution, using Agile disciplines such as DevOps, for technology evolution. It stresses the importance of stakeholder participation for all stakeholders, and of business outcomes over technology.

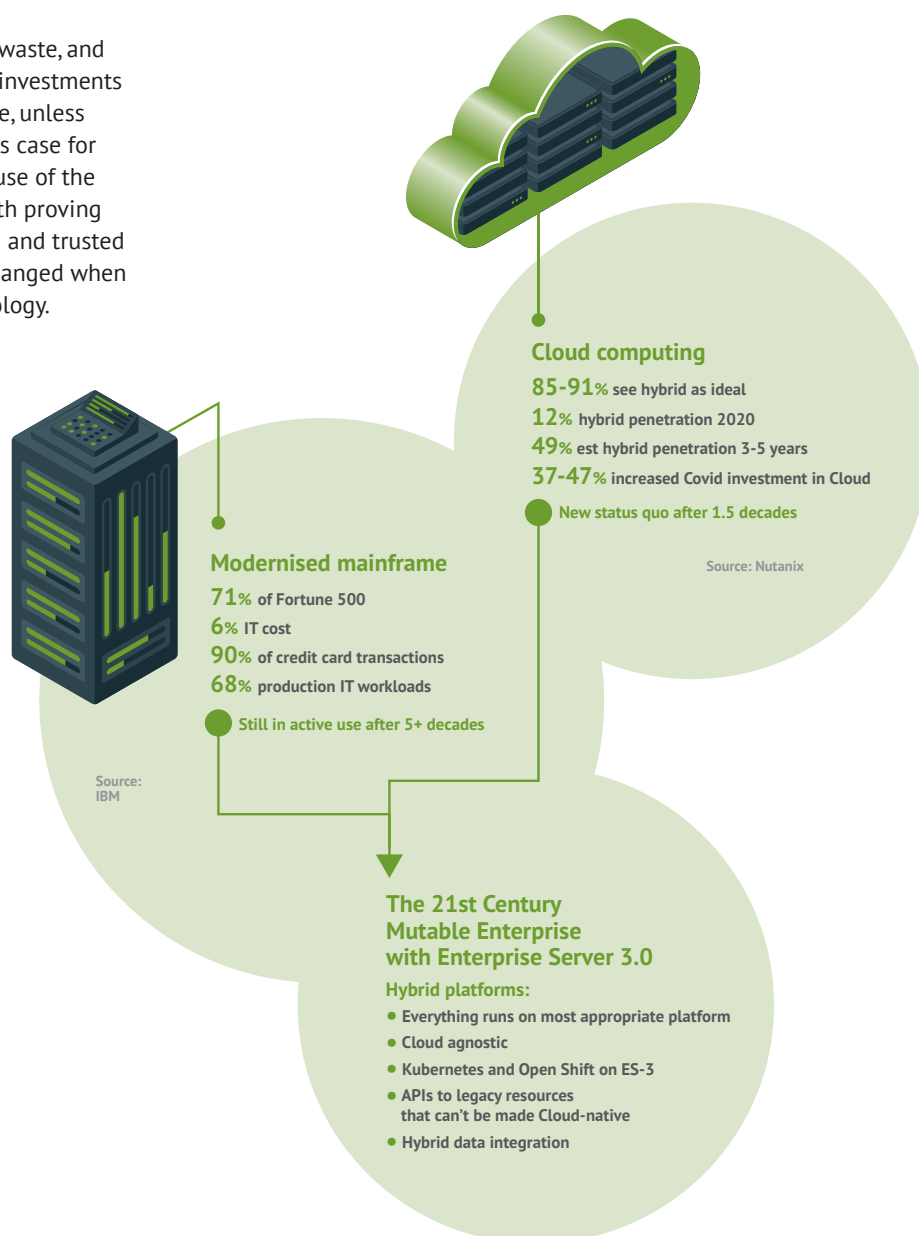
What this means is, that we like new technology and innovation, but we also recognise that whole lifecycle costs matter to the business and that the cost of new technology must be balanced against the value delivered.

In other words, we hate waste, and we hate throwing away old investments that are still delivering value, unless there is a very good business case for doing so. Partly, this is because of the risk and costs associated with proving that the behaviour of an old and trusted (regulated) system hasn't changed when it is rebuilt with new technology.

Luckily, these days, hybrid cloud technology is advancing towards the point where such decisions can be made on purely business grounds, without technology imposing forced constraints. Every workload can, potentially, run on the most appropriate platform for the business needs and this platform can change when or if the characteristics of the business and/or its technology changes.

**Figure 1** illustrates this modernisation process.

**Figure 1 –**  
Modernisation with ES-3



# How do you modernise a working mainframe as ES-3?

“

*We're not just opening up your application to the world, we're opening up the world to your COBOL application.*

”

**Bob Jeffcott,  
a principal  
systems engineer  
at Software AG**

**I**n more technical detail, there are basically 3 ways to begin to bring mainframe legacy applications that still produce value into the mainstream of business automation. Once this legacy has been modernised in this way, it is best thought of as “*just another (but very powerful) server*” – Enterprise Server 3.0 (ES-3).

## **1 – Add bidirectional APIs to legacy applications, without changing the code**

To quote Bob Jeffcott, a principal systems engineer at Software AG: “*We're not just opening up your application to the world, we're opening up the world to your COBOL application*”. Of course, you'll need technology to facilitate this, such as webMethods Mainframe Integration [[here](#)] from Software AG (although there are several alternatives exploiting the API Economy in this space, from other vendors), but there are few technology barriers and little danger of lock-in to any particular vendor.

Nevertheless, there are real issues to deal with, starting with cultural mismatches. Mainframe application support used to be very risk-averse and conscious that change – innovation – could adversely impact the proven value being delivered by the status quo. At the same time, one is assuming that the legacy COBOL, PL/1, Natural or whatever code is reasonably well written, so that its behaviour is well understood – you don't want an API delivering surprises – and it can be maintained, if the business needs new behaviours.

What this means is that even if you are intending to add APIs to existing code, you have to put resources into reducing cultural mismatch and removing silos – don't expect people to do this by themselves and in their own time. Managing cultural change is a whole subject in itself, which we don't have space for here, but we will mention that introducing DevOps is a useful common facilitator, as is rotating staff around potential silos. Training also helps, but is

not sufficient by itself (and learning how COBOL, e.g., works, shouldn't be a problem for a competent developer, but don't count on DIY learning on the cheap).

And remember that this is just the start of a continuing journey, justified at each stage with business use cases. You can progress further on this journey with containerisation, packaging legacy as micro-services, using tools that run on hybrid cloud platforms on both distributed systems and mainframes, to arrive at full ES-3.

## **2 – Unlock Data Value**

“*When data is locked inside systems of record [usually on a mainframe in big enterprises], it's an untapped resource*” (as Danie Bierman, product manager of CONNX at Software AG puts it). There is value to be gained if this resource can be tapped, in real time, by modern cloud-native, containerised applications or cloud analytic platforms.

The good side of this is that ES-3 is usually high quality data, validated, and with the necessary usage permissions – don't throw this quality away by dumping ES-3 data in an unmanaged data lake. Unfortunately, however, the technology mismatch between the world of synchronous databases on, say, IBM zSystems mainframes holding validated and well-structured data and the asynchronous world of cloud data lakes holding data of varying quality, means that cloud applications usually can't just access systems of record directly; and most of the cultural issues mentioned in the prior section apply.

Many tools are available to extract and transform “systems of record” data in real time, although not all of them work across all domains. You need a data integration solution that can translate ASCII encoding to EBCDIC (used on IBM mainframes), enable SQL access (quite a lot of mainframe data may be stored in non-relational databases such as VSAM and IMS) and so on – all without requiring any significant changes to application code.

**Software AG CONNX** is one such tool. Primarily, it has a lot of **database adapters** – over 150 of them – including



such mainframe specialities as Adabas, DB2, VSAM-ESDS and -KSDS, IDMS, IMS, etc. And there's an OLE DB Adapter module for secure, real-time, read/write access to any SQL-based ODBC or OLE DB compliant data source that isn't already covered. Mainframe and mid-range platforms supported include: z/OS, OS/390, MVS, VSE, MVS ESA, z/Linux, AS/400, OpenVMS, VAX, Alpha, Integrity, and Itanium. CONNX is the basis of a [full mainframe data integration platform](#). With the recent addition of [StreamSets](#) to the Software AG portfolio, the combined solution allows for direct connection to these database adapters and pain-free delivery to modern data lake and data cloud platforms – providing a seamless hybrid data integration solution.

CONNX, in essence, allows you to (amongst other things) create a virtual database which is kept up-to-date in real-time, for access from modern business intelligence tools and data driven apps using SQL-friendly access methods such as ODBC, JDBC, OLE DB, etc. StreamSets is able to provide further extensibility and automated delivery of data to schema-agnostic architectures like the data cloud.

Data virtualisation allows you to present a unified view of diverse data sources as one virtualized database, with no need to move the data (although it also provides a staging platform for when you do want to move the data, for replication, ETL etc.). Potentially, this could be a performance overhead, of course, but with good design (only updating what has actually changed, for example) this can be kept minimal, without impacting mainframe performance. In fact, federating data and querying it where it already is may often be the most efficient approach, despite the possibility of network delays when querying multiple sources. As Daniel Howard and Philip Howard point out [\[here\]](#) “query in place... makes a lot of sense when you have good reason to store your data in different places. For example, you might have a number of disparate types of data (structured, unstructured, graph,

documents, time series, sensor data, etcetera) that need to reside in different types of database (and therefore different databases period) for performance reasons. That doesn't – or shouldn't – stop you from wanting to query that data in aggregate, however, which is easy to do with data federation but very tricky to do efficiently using extraction and load.”

Using “virtualised cross-platform database” technology, you can build a virtualised data warehouse on the Cloud – a technology called [Snowflake](#) is a particular enabler for unlocking the value in this data.

While we can easily see that “pay per use” on Cloud, with no on-premises data centre, could make the data in systems of record data more Agile, do we now have “two sources of truth” and two (or more) different environments for the stakeholders in the data to deal with? No, because CONNX virtualises data access and updates in real-time. Snowflake itself is a software-as-a-service data management platform, supporting storage, processing and analytics. It uses its own new SQL query engine and its own cloud-native architecture – it is not built on existing storage solutions such as Hadoop. It aims to offer the data management simplicity of a shared-disk architecture, but with the performance and scale-out benefits of a shared-nothing architecture hybrid of traditional shared-disk and shared-nothing database architectures. Its central data repository holds persisted data that can be accessed from all compute nodes in the platform but it processes queries using MPP (massively parallel processing) compute clusters (each node holds part of the dataset locally), Snowflake specifically promises that:

- There is no hardware (virtual or physical) to select, install, configure, or manage.
- There is virtually no software to install, configure, or manage.
- Ongoing maintenance, management, upgrades, and tuning are handled by Snowflake.

“

**[Snowflake]  
uses its own, new,  
SQL query engine  
and its own  
cloud-native  
architecture –  
it is not built on  
existing storage  
solutions such  
as Hadoop.**

”

Nevertheless, Snowflake cannot be run on private cloud infrastructures (on-premises or hosted), although its data could be sourced from on-premises storage, of course, if you used something like CONNX and Streamsets that supports hybrid data integration. Snowflake provides a thoroughly modern and consistent UX but for a sophisticated cross-platform enterprise solution it won't be just a matter of buying it as a "silver bullet" from the Web with your credit card. It may well disrupt existing databases and warehouse solutions, so (Software AG tells me) you will still need an effective skills-transfer partner to get the best out of sharing data between the mainframe and cloud.

You need a partner, but not one that takes control of your data and uses this to exploit the relationship for its own benefit. Even so, we think there are some issues a company will have to deal with itself, even if its partner can help – you can outsource execution, but not accountability. This is especially the case if data has to cross silo or cultural boundaries, and if a lot of important – regulated – data is being produced and stored on ES-3, the modernised incarnation of the old mainframe. For many organisations, the mainframe was a foreign country, one of the points of ES-3 is that it now isn't, and mustn't be.

There are real technical issues with accessing ES-3 data, although once you've recognised them, they are fairly easy to address. You'll need to think about mismatching security models, different change velocities, identifying (and publicizing) the "single source of truth" for data stored in multiple places. But modern tools deal with this. Once again, the real elephant in the room is cultural change. The custodians of the mainframe data and data custodians on other platforms, the analytics gurus, Security, Compliance Operations must all be talking to each other, with the same goal of delivering Value to the business. No silos. This implies management putting actual budgeted resources into removing silos and generating real collaboration – rather than relying on "collaboration by diktat".

### **3 – Modernising the user experience**

This really merits a section on its own as, for many stakeholders, the user experience is all they see of their business automation. It is all that matters to them – which is why cloud "user experience level agreements" (XLAs) are becoming as, or more, important than the old cloud "Service Level Agreements" (SLAs). In fact, interface modernisation can give you an "early win" to gain acceptance for a bigger transformation process over the longer term. It is important, however, that you are careful to manage expectations. Just modernising the user experience probably doesn't mean that you are anywhere near finished, but sometimes less technical stakeholders see the modern UI as being all that you need.



## Summary: the end of a journey

**D**evelopment of a cost-effective ES-3 platform should be seen as an evolutionary journey, not something you can just do and then forget about. The aim is continual improvement (plan, do, check, act), following the mantra “work smarter, not faster, harder or longer” (although the consequences of working smarter may well include higher productivity, more delivery velocity, and so on). A mutable business can’t afford to waste time and resources on extensive rework, so it will make incremental changes, in units of “minimum viable product” and review them for business impact and input to the next change.

Now, whether you take approach 1 or approach 2 above, or probably both, pay equal attention to also modernising the user experience. Think about providing user-focused web apps that integrate multiple accesses in order to achieve a single business outcome; composite apps using Java or .NET; apps that integrate disparate data sources. And try to deliver small wins as soon as possible, in order to maintain stakeholder commitment to modernisation.

Finally, remember that all stakeholders matter and, with the likely growth of home-working post-pandemic, you can’t assume that you have them locked into whatever on-premises user experience you choose to give them. It is best to keep all of your stakeholders happy – and remember that happy staff usually deliver happy customers.

“

**Remember that  
happy staff  
usually deliver  
happy customers.**

”

### FURTHER INFORMATION

Further information about this subject is available from  
[www.bloorresearch.com/company/software-ag/](http://www.bloorresearch.com/company/software-ag/)



### About the author

**DAVID NORFOLK**

**Practice Leader:**

**Development & Governance**

**D**avid 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 © 2022 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.



Bloor Research International Ltd  
20-22 Wenlock Road  
LONDON N1 7GU  
United Kingdom

tel: **+44 (0)1494 291 992**  
web: **[www.Bloorresearch.com](http://www.Bloorresearch.com)**  
email: **[info@Bloorresearch.com](mailto:info@Bloorresearch.com)**