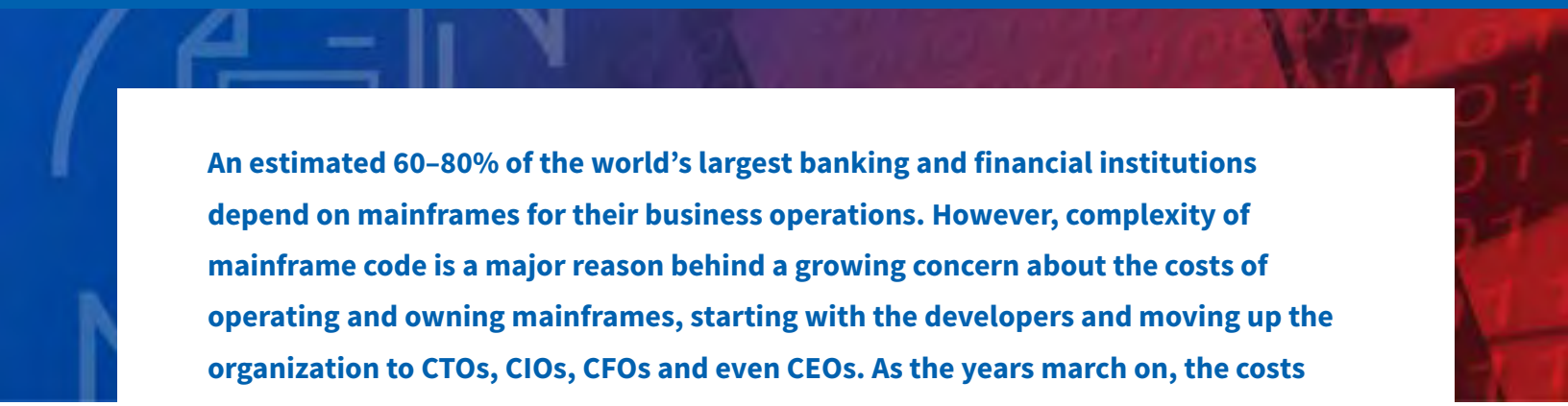
CO_2 Reduction:

Reduce Your Total **C**ost of **O**perations and **O**wnership **(CO_2)**

Paul Bobak, Vice President of Technical Field Services, TmaxSoft Inc.
Narayanan Sundaresan, Vice President, Virtusa Software Services Ltd.



An estimated 60–80% of the world's largest banking and financial institutions depend on mainframes for their business operations. However, complexity of mainframe code is a major reason behind a growing concern about the costs of operating and owning mainframes, starting with the developers and moving up the organization to CTOs, CIOs, CFOs and even CEOs. As the years march on, the costs are increasing dramatically to the point that they are barely sustainable. This paper looks at why this is so and the options available to reduce these costs.

Why is code complexity such a problem?

Over the years, millions of lines of code have been added to mainframes to meet the needs of businesses that are expanding. Yet, in most cases, no one updated the documentation with an explanation of why. And this means that the system has become a black box. Since it's not possible to see what's in this black box, the likelihood that much of the code in the system is redundant is greater. The result is a lot of code that has never been executed in production, creating a dead load that is difficult to pinpoint when people try to troubleshoot or solve poor application performance.

And that brings up another issue: the dwindling number of technical resources with the knowledge to program for, work on and troubleshoot mainframes. Those who do have the knowledge are nearing retirement age while universities and schools are not offering mainframe training of any note anymore. Backfilling positions left open by retirees therefore involves two unattractive options: paying a high price for someone else or making a big investment in training. Yet success is not even guaranteed then. Because the system is a black box, much of the code and applications can be a mystery to even the most qualified mainframe unicorn. Plus, over the years, mergers, acquisitions, and fluctuating markets have left applications without vendors or support.

How does the cost of operation increase?

Think of a mainframe that's been around for 30 years or so. It's been processing an ever-growing number of transactions and storing increasing volumes of data. Many are also running distributed and mobile applications, which forces them to work harder. All of this creates higher MIPS consumption, which raises the costs of operations because most organizations pay fees per MIPS usage. That means a corresponding increase in the overall cost of running the business, often at the expense of being able to offer consumers, customers and other users the fast, modern applications they expect.

How does the cost of ownership increase?

Increased MIPS usage fees are just the beginning. Eventually, more capacity is needed to address the drain on performance caused by high MIPS consumption and the varied workloads that modern applications demand. Additional capacity is available at a price—the price of new hardware and software, followed by the costs for the inevitable upgrades of these investments. Adding capacity or upgrading directly impact TCO.

That impact is bleak. An industry study published almost a decade ago then estimated that the TCO of a mainframe over four years was anywhere from $8 million (for a municipal government) to $23 million (for a financial services firm) with revenues of $4.8 billion in 2010. Not taking into account the massive workloads that have been added to mainframes since 2010 and that have accelerated cost increases, that translates to $9.25 million to $26.5 million in 2018 dollars. Is it any wonder that IT organizations are struggling? This kind of TCO is unsustainable. In addition, there is little, if any, room left to devote to innovative solutions to address modern business needs.

What is to be done about CO_2?

The high costs of the issues caused by aging mainframes and their code complexity has been commonly known for some time. Although mainframe vendors, software companies, and startups have been offering options to reduce CO2 for a decade or more, the advent of the cloud has changed the game and enabled more options. These options can be divided into three main categories.

Packaged Solution (COTS)

Mainframe applications can be replaced with readily available, packaged, commercial-off-the-shelf (COTS) solutions. These solutions offer functionality considered comparable to that of standard mainframe applications; however, they rarely have everything needed for a rapid and easy replacement of complex legacy systems. One answer to this problem is to customize and parameterize the package

"'Lift and shift' is a low-risk, cost-effective and efficient option for reducing the costs of mainframe operation and ownership."

so that it completely matches the existing function of the legacy system, but that requires a significant investment of time and effort. In most cases, companies decide to change business processes to accommodate the packaged solution—even if that means retraining or additional expenditures. This might appear to be a fair tradeoff; however, these processes are often part of a business's unique value proposition, and accommodating the solution puts that value proposition at risk.

Application Rewrite

Another option often considered is to rewrite mainframe applications on a modern, yet lower-cost platform, like opensystems on premise or in the cloud. Automated code analysis, code conversion, testing, and cloud deployment tools are used to facilitate what is, in most cases, a significant reengineering project. If all goes well, a company can end up with the most advanced application available, but at what cost, time and risk?

Code conversion is often inaccurate and unsatisfactory with few guarantees that it will be operable. Meanwhile, rewriting complex applications with millions of lines of code includes redeveloping business logic and translating databases, and it's nearly impossible to get all this done in a short amount of time. The errors encountered in automated code conversion are so time consuming that it could take almost the same amount of time as a manual rewrite. In fact, it may be years before the rewrite is launched in the newer platform and all the while, your mainframe is still running and incurring the same costs as before. Moreover, understanding the existing functionality is critical while making sure that nothing is lost in the entire exercise of conversion. If something is lost, it might not be easy to trace, disrupting the project and even the business.

"Lift and Shift'" aka Rehosting

"Lift and shift" is a low-risk, cost-effective and efficient option for reducing the costs of mainframe operation and ownership. Mainframe applications are rehosted as-is on a lower cost platform with minimal or no change to the application code, business logic or user interface. Services equivalent to those provided by the mainframe are run from the new platform, and the process can take as little as nine months.

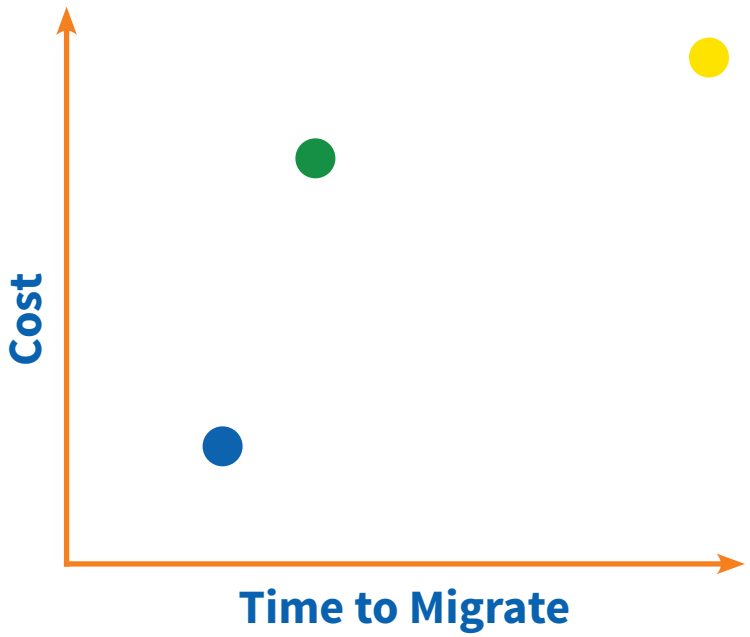
The lift-and-shift option ensures that the applications are still operating in a mainframe context. Some industry observers see this as a disadvantage, pointing out that, at the end of the day, organizations still must maintain brittle application code. Most organizations that have rehosted, however, report few issues. Some organizations take a careful

hybrid approach by moving some of their least critical applications first and then move the other applications after making sure that this option works well and is stable.

Which option is the best?

When there is a legacy modernization dilemma, organizations that choose to remain on the same platform by doing nothing or who decide to upgrade their mainframe system will continue to deal with mounting operation and ownership costs apart from being left far behind on their digital transformation journey which will leave them answerable to business stakeholders and CXO's internally. Therefore, neither is a good option unless an enterprise has millions of dollars to burn and doesn't mind existing with systems from a forgotten era.

So, it's down to rewriting, rehosting and hybrid "lift-and-shift." This illustration shows the cost benefits and migration times of the three remaining options. This is just an illustration and may vary per actual solution.



- **Rehosting:** complete lifting and shifting away mainframe applications
- **Hybrid:** move some mainframe applications now, move others later in a phased manner
- **Rewrite:** completely rewrite and modernize the applications currently on the mainframe

Application rewriting vs. rehosting

Application rewriting scores highest for benefits. After all, the application is being built from the ground up with the maximum flexibility of the digital era, using the latest cutting-edge technology. But that does not necessarily mean it is the best option. In the illustration, you can see that it is almost “off the charts” when it comes to migration time. Understanding the existing code and functionality and to rebuild the entire code base in the newer platform are not things that can take a few months. Also, the risk of reverse engineering the entire codebase and losing functionality in transit is great. In fact, not only is the margin of error high, but so are the odds of business disruption should something be missed and the application fails.

For risk-averse organizations who want to modernize applications by migrating, re-hosting is the better option to pursue. Existing business logic is untouched, and the entire code base gets shifted to an open platform. Compared to an application rewrite:

- The risks and costs are low.
- The difference in migration time is dramatic.
- No special user or resource training is required.
- Code conversion is minimal.
- Mainframe TCO is reduced more quickly because capacity moves off the mainframe in what can be less than a year, which means lower hardware and licensing costs.
- MIPS consumption drops much more quickly than with rewrite because of the faster migration, lowering operation costs

Rehosting is also the best interim option for customers to move their existing applications off the mainframe to reap the benefits of lower operating costs, and the savings can fund part or all of the rewriting process.

Hybrid “lift-and-shift”: A way to rule them all

The hybrid method is a more deliberate form of rehosting, offering even lower risks than application rewriting or a comprehensive rehosting project. The process migrates apps necessary for running the business on a new open platform, but they are chosen by determining how much impact moving them will have on business processes. The less impact, the better.

Some organizations choose the hybrid method to migrate aging legacy applications whose original owners and operating details are most likely lost. Because the systems they rely on are often hidden behind the misty

walls of obsolescence, these apps drag down mainframe performance. After a few of these apps are migrated with success, organizations can begin to move over their most mission-critical apps. Meanwhile, costs are already lower because the apps impacting the mainframe performance are gone.

Virtusa and TmaxSoft, have an approach that can be applied to rehosting or hybrid “lift-and-shift.”

The Virtusa and TmaxSoft Approach

Virtusa approaches rehosting with a plan for reducing technical debt by cutting the operations cost and TCO of running mainframe applications with a shorter recovery period. The strong partnership between Virtusa and TmaxSoft enables this achievement. We use the following five-step process to successfully achieve this migration.

Step 1: Analyze

Before migration, we analyze the entire set of code to be rehosted. We isolate this application code from the present system, while looking for any external dependencies or linkages.

Step 2: Capture

We capture the code base in a tool to understand its complexity based on industry standard parameters such as Halstead, McCabe, McClure, essential complexity and function points. Another objective of this step is to reduce the technical debt before migration by identifying dead and redundant code. We use exclusive partner tools for this cleaning process.

Step 3: Migrate

In this step, the new environment is prepared for migration and all the code is moved from the mainframe. The migration is done in three separate phases: database, application and file or dataset migration.

Step 4: Check

After the migration step is complete, we check and review the entire process to ensure the new system is performing well and is stable. If we find anything that does not meet the optimum performance level, the process goes back to step 2 and we start over.

Step 5: Go live

This is the final step. We bring the new system to production and turn off those applications on the mainframe.

Conclusion

Mainframes continue to run core business applications for most banks, other financial institutions and the Fortune 500. Although they were given a "death sentence" a decade ago by analysts, they live on. Unfortunately, many are on life support, and keeping them there is costing companies millions of dollars in operations and TCO. Of the options available to reduce the costs of operations and ownership, rehosting is the most cost effective because it doesn't require years of programming and reengineering, and the applications most affecting performance and driving up costs can be moved in as little as nine months. If you are considering reducing your mainframe's CO2, it is the first option to evaluate.

virtusa®
Accelerating Business Outcomes

About Virtusa

At Virtusa, we accelerate business outcomes for our clients through our expert information technology consulting and outsourcing services. We support a wide variety of Forbes Global 2000 firms with services that span the entire spectrum of the IT services lifecycle. Our industry-leading solutions transform businesses not only for a better today, but also for a better future. Listen to our podcast series to find out how your business can benefit from emerging technologies to become a truly intelligent enterprise.

Virtusa Corporation
132 Turnpike Road, Suite 300
Southborough, MA 01772
TEL: +1 508 389 7300
www.virtusa.com

TmaxSoft

About TmaxSoft

TmaxSoft was founded on the principle that there is always a better way. Nothing will stop us from creating better solutions that improve your business in ways that other companies can't. Or won't. A global software innovator focused on cloud, infrastructure and legacy modernization, we give you the freedom to take control of your technology. Our complete stack of software solutions help you efficiently manage and fully leverage your critical data. For more information about rehosting, download our new eBook, *Lift, shift and modernize: proven mainframe modernization strategies that enable digital transformation*.

TmaxSoft

GLOBAL HQ

TmaxSoft, Inc.

230 W. Monroe St., Ste. 1950
Chicago, Illinois 60606

TEL : +1.312.525.8330
Email : info@tmaxsoft.com
tmaxsoft.com

Korea Group HQ and R&D Centre

TmaxTower 45, Jeongjail-ro,
Bundang-gu, Seongnam-si,
Gyeonggi-do, Korea 13613

TEL : +82.31.8018.1000

For all office locations, visit www.tmaxsoft.com/about/locations