IT'S ALIVE!

AVOIDING THE

FRANKENSTEIN
SYNDROME

IN ENTERPRISE IT

BY **MARK NEFT**
EPAM United States
October 2017

Table of Contents

THE DRIVE TOWARD LEGACY MODERNIZATION IS CREATING ENTERPRISE IT FRANKENSTEINS

"As the minuteness of the parts formed a great hindrance to my speed, I resolved, contrary to my first intention, to make the being of a gigantic stature, that is to say, about eight feet in height, and proportionably large. After having formed this determination and having spent some months in successfully collecting and arranging my materials, I began."

MARY SHELLEY, FRANKENSTEIN

Despite Victor Frankenstein's months of careful planning to build his creature, it still became a monster. In the world of enterprise IT, even though stakeholders from across the business have the best intentions, a similar Frankenstein Syndrome is happening when it comes to legacy application modernization. Indeed, the pursuit of cost savings and best-in-class solutions has led some enterprises to build their own monsters comprised of many moving parts that, like it or not, just don't fit together.

This phenomenon begs the questions of whether or not enterprises know what they already have with their legacy applications and, specifically, those that run on the mainframe. With worldwide spending on enterprise application software predicted to grow from $149.9 billion in 2015 to $201 billion in 2019,[1] businesses need to modernize their legacy enterprise applications with a holistic, big-picture approach to ensure they're not building a Frankenstein's monster that will be difficult to maintain in the future.

With the expected growth in enterprise application spending, there are many potentially unforeseen risks associated with achieving enterprise IT excellence. This white paper will show how many of these risks can be mitigated by understanding the implications to the organization from four perspectives: Network, Operations & Management, Development Platforms and Business Continuity. By exploring these perspectives, we can become more aware of the big picture and avoid building our own enterprise IT version of Frankenstein's monster.

---

[1] http://www.gartner.com/newsroom/id/3119717

MAINFRAME OPTIMIZATION CURES THE FRANKENSTEIN SYNDROME

"You are my creator, but I am your master; Obey!"

MARY SHELLEY, FRANKENSTEIN

Many organizations that have mainframes installed today have been inadvertently affected by the Frankenstein Syndrome. To accomplish this feat, organizations have spent several years of careful planning and execution trying to reach the future state. Creating Frankenstein's monster was not the end goal, but more so a byproduct of chasing unrealized cost savings and moving toward innovative solutions that simply don't work together.

Why has this happened? Many organizations have been myopic in creating future-state architecture. They have just been focusing on cost reduction or the shiniest, newest toy. What organizations end up with is a solution that looks and acts like Frankenstein's monster – one that could eventually destroy the organization. Even with the best intentions, this will continue to happen until organizations understand their current platforms, and how the mainframe and its ecosystem can not only save the organization money, but can also reduce the overall complexity of enterprise business solutions.

We believe this phenomenon is happening for two primary reasons:

Organizations are not thinking of things in terms of application ecosystems or how the applications work together

These organizations have not been keeping up with platform updates, such as on the mainframe, or don't know how to leverage the newer capabilities of existing platforms

In either case, organizations can avoid creating their own version of Frankenstein's monster with a little knowledge and planning.

HOW TO AVOID THE FRANKENSTEIN SYNDROME FROM FOUR PERSPECTIVES

"A mind of moderate capacity which closely pursues one study must infallibly arrive at great proficiency in that study."

MARY SHELLEY, FRANKENSTEIN

According to a 2015 Gartner survey,[2] "45 percent of respondents with knowledge of their organization's software strategy indicated that one of the current top five IT project priorities is 'application modernization of installed on-premises core enterprise applications' and a further 41 percent indicated that 'extending capabilities of core enterprise applications' is a top five priority." Two years later, there's no doubt that legacy (and mainframe) modernization is still top-of-mind for any organization who hasn't tackled the challenge yet.

With so much emphasis on enterprise IT and applications as a priority, we should learn how to avoid common pitfalls of legacy modernization by studying the characteristics of Frankenstein's monster. We can do this by looking at just the four primary factors that contribute to the Frankenstein Syndrome in enterprise IT:

NETWORK: The amount of data moving across the network	**OPERATIONS & MANAGEMENT:** The number of different physical/virtual technologies that have to be managed as one
DEVELOPMENT PLATFORMS: The number of different application architectures	**BUSINESS CONTINUITY:** The amount of extra physical infrastructure required to support availability requirements

If organizations understand how the modern mainframe can address these four critical aspects of an enterprise solution, then there will be fewer instances of Frankenstein's monster popping up at the enterprise door.

---

[2] http://www.gartner.com/newsroom/id/3119717

PERSPECTIVE I: NETWORK

As organizations chase cost reduction, the first thing they might attempt is to move processing off the mainframe without moving the data as well. On paper, this move might seem as simple as drawing a line connecting the new world to the old world without considering the three primary areas of limitation that the network imposes on enterprise solutions: Speed, Width and Security.

With today's routers, switches and so on, network speed has been solved as far as physics will allow. Data can move from one place to another at the speed of light.

The second consideration may be a little harder to understand, but we're referring to the width of the network between two points. For example, if the application is moving a small amount of data across the network, then this will have little impact, but if there is a large amount of data moving across the network then this will have a very big impact.

This can be illustrated by using the analogy of moving a large number of people from one place to another. Imagine you only have a Porsche that can move one person at a time on a high-speed road. On the other hand, there is a bus that can move 44 people at a time. This would be the difference of a standard MTU of 1,500 and an MTU of 65K that you might see in a physical network connection between two environments. In most data centers, servers are connected together through switches that force an MTU of 1,500.


This image illustrates the difference of moving one person at a time vs. a bus load of people at a time; what do you want within your network when you need to move large amounts of data around?

The final network consideration is security. As the application ecosystem expands across a physical network, the need for encryption emerges. So, an architecture that enables ubiquitous processing also needs to enable ubiquitous security. Considering the amount of encrypted data that is processed, the encrypt/unencrypt process could contribute to application latency.

PERSPECTIVE II: OPERATIONS & MANAGEMENT

When organizations introduce ubiquitous computing on and off the mainframe, they also need to handle the operations and management of these new physical and virtual platforms, especially if the application ecosystem is extended to the cloud.

As more and more distinct and emerging technologies are glued together to create best-in-class computing solutions, the operations aspects of managing the new environment can become an N-factorial type of problem based on the number of technologies being managed, assuming they all need to talk to each other. Introducing more and more platforms to this scenario makes it impossible for the organization to ever achieve Agile IT. With as little as seven different platforms all talking to each other, the Operations team now has to manage 28 unique interfaces compared to the single platform of the mainframe.
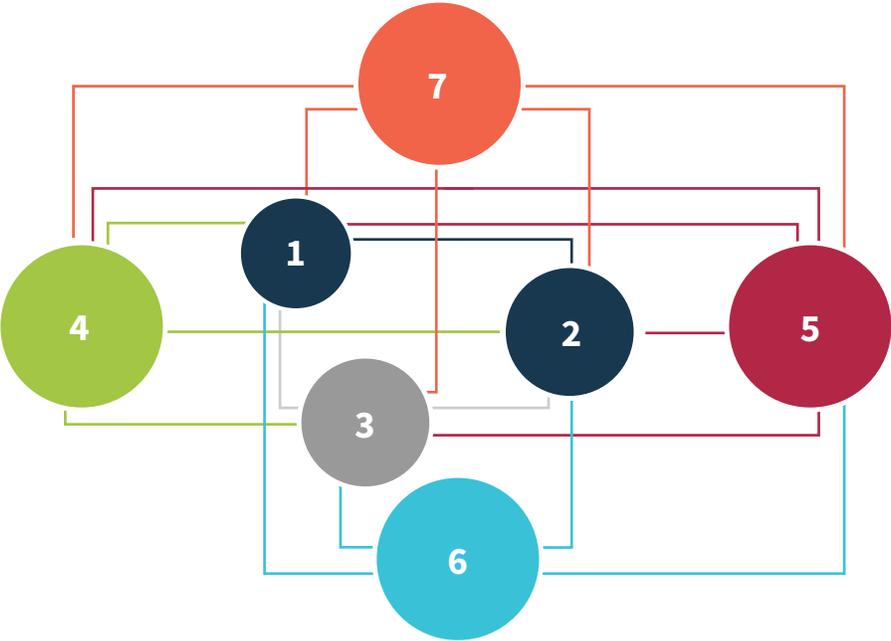
AS NODES GET ADDED, THE TOTAL NUMBER OF MANAGEMENT POINTS INCREASES
BY N CREATING AN N-FACTORIAL MANAGEMENT PROBLEM

NUMBER OF NODES	POINTS OF MANAGEMENT
1	1
2	3
3	6
4	10
5	15
6	21
7	28

PERSPECTIVE III: DEVELOPMENT PLATFORMS

As organizations chase lowered costs and increased productivity, they look to new development platforms for the right answer, but have they looked at exactly what makes the new development platform less costly and more productive? In many cases, the lower costs can be contributed to the ability of running on Linux regardless of the chip architecture. The productivity is typically a byproduct of a modern Integrated Development Environment (IDE) and a flexible framework that reduces the amount of code that needs to be written to create applications and the ability to standardize actions.

The questions that must be asked before moving to the new development platform are:

- 1 -	- 2 -	- 3 -
Is there an IDE that could enable additional productivity for my existing developers?	Is there a flexible framework that reduces the amount of code that needs to be written for my existing codebase?	Is there a flexible framework that can help ensure application standardization?

The answer to the first question is yes, and there is more than one option when it comes to IDEs. The second question is a little harder to answer because it is based on the current coding language. There is a different answer if the current language is assembler compared to if it's COBOL. For example, there is no framework for a mainframe assembler, but there is a new modern framework for COBOL.

So when looking at moving development from COBOL to Java, Ruby, Python, etc., it's useful to take a step back and look at the existing applications within the organization and ask yourself the following question: How long did it take my COBOL application to become unmanageable as opposed to my Java application?

PERSPECTIVE IV: BUSINESS CONTINUITY

The possibility of failing because of a complex application ecosystem that is made up of mainframes, distributed x86 servers, and an external cloud from one location to another is a nontrivial problem to solve. Many organizations have plans on paper, but hope they never have to execute them since they don't know what will happen or even if everything will work after execution.

As organizations extend their processing off the mainframe and into an extended processing model, the amount of excess capacity needed instantly doubles to handle the required availability. It does not matter if the organization is using an active/active or active/standby deployment model – the amount of excess hardware is still 100 percent to enable full production failover at peak capacity.

One of the best kept secrets in the mainframe space is the ability to have a standby mainframe with minimal capacity turned on to maintain operations on the standby machine. This capability enables the organization to have a Disaster Recovery mainframe with a minimal configuration, thus lowering overall software and hardware costs.

FOUR STEPS TO CURE THE FRANKENSTEIN SYNDROME

In addition to the four we already discussed, there are many other contributing factors to Frankenstein Syndrome that organizations should consider, such as development technologies, mobility and user interfaces, just to name a few. Keeping all of these factors in mind, there are four simple and affordable steps that every organization can take to prevent Frankenstein Syndrome, or even prevent falling victim to it in the first place:

1   **Be honest with themselves** about what they want to achieve and what they have today. I see that many organizations make platform decisions based on emotions vs. facts, or they assume that the change to a new platform will solve the perceived problems.

2   **Know the true baseline costs.** Many organizations think that mainframes cost too much because it is the largest single line item on the budget, but they don't consider the number of applications being supported or the amount of business being supported. Typically, when the costs are normalized against the book of business, the costs are lower than perceived.

3   **Know what the organization does well.** Many organizations don't realize that the mainframe is the only platform within the organization that provides the 24x7x365 availability required for key mission-critical applications.

4   **Get educated on the "Art of the Possible"** on a modern mainframe. Businesses might be surprised by what they can do today and how it can enable a modernization journey with minimal risk.

Frankenstein Syndrome can be avoided by taking proactive measures, providing early care and feeding, and, most importantly, channeling an aspirational vision that brings the entire organization along for the journey instead of hiding the monster – the mainframe – in the dungeon of the organization. Ensure that when building this aspirational vision, decisions are made based on facts, not emotions, and that you don't treat your applications like your children or disciples, but rather as pillars capable of acting as a foundation to support the organization.

ABOUT EPAM SYSTEMS

Since 1993, EPAM Systems, Inc. (NYSE: EPAM), has leveraged its core engineering expertise to become a leading global product development and digital platform engineering services company. Through its 'Engineering DNA' and innovative strategy, consulting, and design capabilities, EPAM works in collaboration with its customers to deliver innovative solutions that turn complex business challenges into real business opportunities. EPAM's global teams serve customers in over 25 countries across North America, Europe, Asia and Australia. EPAM is a recognized market leader among independent research agencies and was ranked #12 in FORBES 25 Fastest Growing Public Tech Companies, as a top information technology services company on FORTUNE'S 100 Fastest Growing Companies, and as a top UK Digital Design & Build Agency.

LEARN MORE AT **HTTP://WWW.EPAM.COM/** AND FOLLOW US ON TWITTER @EPAMSYSTEMS AND LINKEDIN.

GLOBAL

41 University Drive, Suite 202
Newtown, PA 18940, USA

P: +1-267-759-9000
F: +1-267-759-8989

SALES CONTACTS

Please contact us for any sales inquiries or general information on EPAM.

ENQUIRIES@EPAM.COM