

Building a Compendium For Modernization of Computer Applications

By Joseph Gulla, Ph.D.

Building a Compendium For Modernization of Computer Applications	1
Abstract	1
Strategies, Methods and Toolsets for Modernization.....	2
Organization Matters.....	2
My Motivation to Focus on Application Modernization	2
Changes to the Code Versus Adding a Layer of Functions	2
Why Modernize?.....	3
Database Technology.....	3
Mobility.....	3
Internetworking.....	3
Sources and Methods.....	3
What's a Strategy and How Is it developed?	4
Thought Leaders.....	4
Simple but Useful.....	4
Many Others are in the Mix.....	5
Application Modernization Methods: What's Useful?	5
Method Examples.....	6
Checklists and Frameworks	6
A Checklist for Successful Modernization.....	7
System Assessment Framework:.....	7
IBM Services for Application Modernization	7
Proposal-Led Modernization	7
Application Modernization and Toolsets.....	8
Toolsets as Part of Services	8
Toolsets Supporting Strategy Work.....	8
Templates Derived From the Best Examples	8
Toolsets Supporting Methods Work	8
Harvesting Intellectual Capital.....	9
A Better Knowledge Network	9
Toolsets via Product Features	9
Beyond Strategies, Methods and Toolsets for Modernization.....	10
References.....	11
About the Author.....	12

Abstract

In this paper, the author provides a survey of aspects of the modernization of computer applications. Specifically, the focus of the paper is modernization strategies (plan of action, policy or blueprint), methods (specific procedure, systematic or established pattern) and toolsets (programs to support tasks from analysis and planning to source-code conversion). The scope of the topic is both traditional modernization (changing the source code) and other approaches using interfaces like APIs to access the function and data with the goal of doing something new and different with the applications without changing them.

Strategies, Methods and Toolsets for Modernization

The study of application modernization deserves a systematic and organized approach because of the size and complexity of the topic. In one-way or another, organizations have been modernizing since right after the first application ever written went into production. What systematic approach fits best to understand this decades long pursuit?

Organization Matters

A useful approach to the topic is to organize the ideas with a focus on strategies, methods and a toolset. Strategies are like a plan of action, policies or blueprints. Methods are specific procedures or systematic ways of approaching modernization. And a toolset might consist of procedures, programs, views, schema and data in support of the task of updating or replacing an application. The toolset elements work together to support the modernization task.

My Motivation to Focus on Application Modernization

When I started as a programmer, we were replacing manual procedures with a computer application that made people more productive and reduced errors while providing faster turnaround of information and reports. There was no previous application to modernize as we were creating the first generation of applications. In my situation, after a few years of application programming, I moved on to other jobs in IT and only returned to application development after more than a 35-year absence. Why did I return?

I returned because I had a brief special project where the focus was two major long-standing applications. The goal of the project was to help figure out how to extract the main elements of the application's business functions so that they could be put into a new replacement application. Generally, the task was about extracting business rules and putting them into a repository. The applications I examined seemed frozen in time. The applications were written in COBOL using VSAM and CICS commands supported by basic debugging tools. It was like I was back at the beginning of my IT career. I wondered how these applications could have survived decades seemingly without upgrades of any kind.

Changes to the Code Versus Adding a Layer of Functions

Shortly after my project with the two major COBOL applications, I began to work part time on a project doing research and writing for an API management software company. This company provided a tool to modernize applications from the outside using automatically created Java programs to unlock data and legacy application functionality without changing the application itself. This is an innovative idea, particularly powerful because of the challenges associate with changing existing applications that have grown and evolved over many years. This idea is now an accepted strategy for application modernization.

These two experiences with application related projects were the beginning of my interest in application modernization.

Why Modernize?

Before we explore strategies, methods and toolsets for modernization, let's consider motivations. What drives the need for application modernization? No doubt, technology change is a major motivation. Consider just three factors.

Database Technology

Prior to the creation of database technology, there were access methods. Conventional access methods, particularly those supporting sequential and indexed access, were used in batch and online applications. When database technology emerged, IT departments quickly saw the benefits of putting key organizational data into databases but hurdles existed in doing so. Using database technology required training, additional computer resources and more of a focus on the business aspects of the applications like the creation and use of data dictionaries. Many organizations overcame these challenges driven by the many benefits of database use but other organizations struggled and some never made the transition.

Mobility

The use of mobile devices in everyday life is setting new expectations for organizations' business applications. Simply put, there is a need to provide at least some application functionality on mobile devices like tablets and smartphones. If an organization can overcome the technical challenges of uplifting existing applications, the benefits can be significant to both employees and customers.

Internetworking

The widespread availability of network connectivity has provided many opportunities for applications to be used in ways and situations previously unobtainable. In years past, the desktop device was not a computer but a fixed function display device. A computer that could emulate the fixed function display as well as run desktop applications replaced this display device. Those computers, when connected to an IP network, evolved quickly, supporting Internet protocols and an enormous variety of network-dependent applications running on the World Wide Web.

It's important to consider these factors as drivers for application modernization as well as the emergence of new and powerful programming languages and innovative approaches to developing and sustaining applications.

Sources and Methods

Where can you find out about modernization strategies? How do you survey them? When I was in college, I learned about statistical significance and sample size. When you do a study, even an informal one, it's necessary to gather a large enough sample so that the data you discover can inform your work. If you don't sample, you're just giving your opinion or insights, which isn't so bad if you're an expert in the field. I am not a modernization expert, so I'm using a sample of more than 30 documents (papers, articles, blog posts and websites) from diverse sources (consultants, bloggers, software developers, technical consultants and marketers) to inform my work.

What's a Strategy and How Is it developed?

A strategy is a policy that describes the plan to move forward. In this case, the strategy describes the way that the handling of an application's modernization is going to play out. Think of the general strategy as a blue print or game plan. As you can imagine, there are a number of different ways that organizations express their strategy, which are likely tailored to the needs of a company so there is a cultural fit.

It's an organization's responsibility to develop its strategy about application modernization. Often, this is just the kind of challenge that requires assistance to get started so organizations get help to figure it out. Who can help? Companies like Accenture and Deloitte are often consulted for their industry expertise. Just reading their papers from the web is a big help.

Thought Leaders

Some consultants make a living by being strong in technology and another area like financial services. This analysis, [Modernizing Transaction Banking \(2017\)](#) is a good example of how the intersection of banking and technology plays out in leadership briefs from Deloitte.

In addition to the industry-focused consultants, there are technology and research firms like Gartner, Forrester and IDC that develop thought leadership in many areas including application modernization. Even the simplest materials on the web from Gartner are stimulating and useful in helping to shape a strategy.

Simple but Useful

In the Smarter with Gartner post, [7 Options to Modernize Legacy Systems \(2018\)](#), contributor Susan Moore writes, "If you need to modernize legacy applications, the best approach depends on the problem you're trying to solve" and "Replacement isn't the only option."

If your job is to help formulate a modernization strategy, then an immediate response to reading this is, "What problem are we trying to solve?" You're likely to discover that there are several different problems and perhaps different solution approaches depending on the problem. Moore lists seven different modernization approaches and explains them in a sentence or two. The list includes encapsulate, rehost, replatform, refactor, rearchitect, rebuild and replace.

There are so many factors in formulating a modernization strategy that require careful thought and analysis. Even when care is given to the issues of business drivers, corporate culture, existing skills, use of strategic or technical consultants and time and money, events can undercut the best-developed strategy. Organization change, which is frequent at the leadership level in many IT organizations, can cause starts and stops in carrying out strategic initiatives. These disruptions typically undercut the ability of IT to have a sustained strategic effect on the operations of the organization.

Many Others are in the Mix

It isn't just industry consultants and technology researchers that drive application modernization strategy. Many organizations have trust relationships with technology companies and service providers and they leverage these relationships to develop strategies that are focused on technology and service solutions.

It's easy to imagine a two-column table where Column 1 has a list of problems and Column 2 contains the suggestion of a tactic, product or service that will address that problem. This is a tempting way to approach an application-modernization strategy—the non-strategy strategy. For IT departments that have trouble just keeping up with fixes and basic changes, the notion of developing a detailed application modernization strategy may not be a good fit. What might a sample two-column table look like?

Table 1. Challenges and Solutions

Modernization Challenge	General Solution
Users want to have flexibility to query data collected by applications.	Migrate key indexed files to a database and provide prebuilt data queries.
Users want to use applications from the web and from mobile devices.	Provide web interfaces that can be used on desktop computers and mobile devices.
Programmers want more advanced capabilities in the program languages they use.	Update existing languages and make available advanced functions that were not previously implemented.
Users want the flexibility to manage business rules in a way that supports changes needed to better compete in the competitive marketplace.	Implement software that places business rules in a repository that can be managed by the business community.
Users want changes to the application to be made available on a real-time not daily or weekly basis.	Implement tools and methodology that support continuous development and integration.
Programmers want support for a broader selection of programming languages for databases used with their applications.	Install and enable the latest level of middleware, for example, Db2 with support for languages like REXX, PL/I, COBOL, RPG, Fortran, C++, C, Delphi, .NET CLI, Java, Python, Perl, PHP, Ruby and others.
Application designers want the latest capabilities of their transaction server to build cloud-enabled applications.	Install and enable the latest releases of transaction server software, for example, CICS provides cloud support, performance optimizations, enhanced metrics, additional security, and DevOps engagement to automate CICS deployments.

Application Modernization Methods: What's Useful?

The starting point for any methods discussion is “What is it for?” because the method will vary depending on the modernization task at hand. Is the task small and simple or large and complex? Is the effort a programming enhancement or something much bigger like a

new release of the application? The change to the application could rely on the addition of middleware or an upgrade to an existing piece of supporting software. The variability in the inputs, transformation and outputs is something that I have written about recently in the article *Modernization Is an Ongoing Strategic Activity* (Gulla, 2019). You'll find this figure in the section called "Is There an Application Modernization Method?"

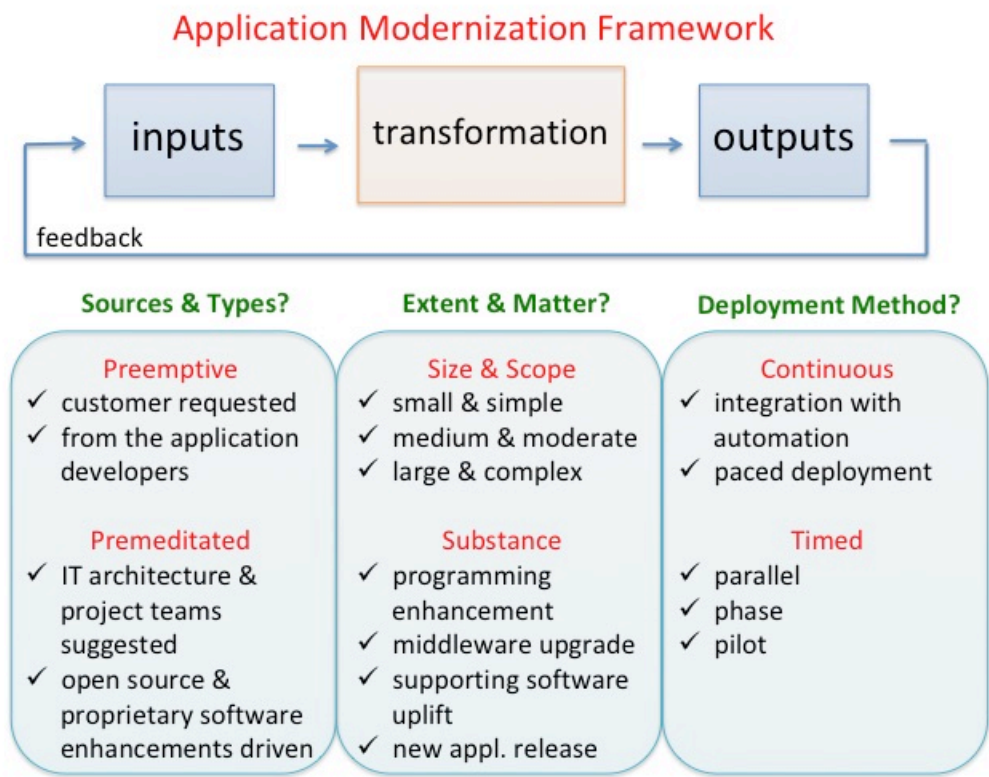


Figure 1. Is There an Application Modernization Method?

This figure lists the variables you should consider when implementing a change to an application. It doesn't suggest a specific procedure to follow, so what methods are out there for use?

Method Examples

Different companies—research or software or professional services—have diverse approaches to the question of methods. Since methods are the bridge between the strategy and project plan for the work, there is the potential for a lot of variability. A tools and services company like might simplify the methods question by getting right to the modernization tasks. This would include GUIs/webification, switching to relational databases, lift-and-shift, SOAization, rearchitecting and automated test case construction/execution. The idea is to pick one or more of these tasks and put together a plan. Of course, there would be some documented background on why specific modernization tasks were needed.

Checklists and Frameworks

Software engineering companies like Altexsoft have "how to" whitepapers like *Legacy System Modernization: How to Transform the Enterprise for Digital Future* (nd) that

contain checklists and frameworks that are part of their method of doing the work. For example:

A Checklist for Successful Modernization

- ✓ Assess the current state of legacy systems
- ✓ Select the modernization approach that would be the fastest to deliver value
- ✓ Rethink the architecture and prioritize for simplicity
- ✓ Choose the technology stack to deliver optimal performance and user experience
- ✓ Document for future system growth
- ✓ Create a separate support and retirement schedule for your legacy system
- ✓ Budget for training and system updates

System Assessment Framework:

- Technologies analysis
- Architecture audit
- Code review
- UI/UX review
- Performance testing
- Current requirements and opportunities for future growth

IBM Services for Application Modernization

The readily available material from IBM lists modernization topics that others don't list, such as AI, automation and blockchain. They also focus on agile, DevOps, microservices and APIs so their emphasis is to help you make leading-edge change.

The services they provide contain the methods they use. For example, Rationalize the Applications You Have (nd) looks at which apps can be upgraded, sunset or moved to the cloud. That service uses a consulting method. IBM also provides a variety of Application Services (nd) like custom application development and management, as well as application automation services.

Proposal-Led Modernization

Some organizations hire companies to guide and handle their modernization challenge. In Modernization: Clearing a Path to Success (2010) the authors explain the process they followed for a client. The steps were:

1. Identify the modernization problem to solve and the opportunities solving it would bring
2. Categorize the approaches, for example, rewrite the application versus buy a commercial off the shelf (COTS) solution versus modernize what they already have
3. Get bids from suppliers for the different modernization approaches
4. Make an informed choice based on the success rates of different projects by type

In general, the authors wrote favorably about the approach to “modernization what you have” versus rewrite the application or the COTS solution approach. They write “For

better or worse, the users get what they had before the modernization project, but improved and less costly.”

Application Modernization and Toolsets

Toolsets in support of modernization have a surprisingly large role to play in any organization’s modernization project or program. In general, toolsets contain programs to support tasks from analysis and planning to source-code conversion. But examined more closely, many toolsets contain procedures, data and skilled human support to help them take on an even broader and important role.

Toolsets as Part of Services

360 Quadrants provides an application modernization services quadrant that compares 34 vendors in application modernization services across 73 criteria. Included in their evaluation is an evaluation of service tools. IBM, Accenture and HCL finished 1, 2 and 3 in the latest evaluation ([Application Modernization Services](#), 2019).

Toolsets Supporting Strategy Work

When consultants help an organization formulate a modernization strategy, they typically call on the work that they have done for other customers.

The most effective consultants maintain a database for their past engagements that contains artifacts like proposals, contracts, project plans, status reports, project reports and metrics like start and end dates, project risk and remediation reports. When you collect this kind of data, you can use it in so many ways, from improving your estimates to sharing information about your project success and reusing materials that are common to all modernization projects.

Templates Derived From the Best Examples

For consulting organizations that purposely resist reuse of materials created for previous customers, templates are the answer. Templates, with the purpose of creating a certain level of consistency and content quality, are used to provide a starting point of materials for every project. Busy consultants appreciate getting this kind of help. They can be provided with:

- Proposal templates that included problems to be addressed, as well as typical costs and benefits of the project
- Project plan templates that list major milestones for the project, as well as details necessary to effectively manage the implementation of the strategy work and other types of projects
- Project report templates that contain sections, heading and subheadings likely to appear in the typical customer’s report

Toolsets Supporting Methods Work

Toolsets play an even bigger role when it comes to the method used to carry out the project. All the toolset examples for strategy work, like proposals and project plans, apply to methods, but many other tools apply as well. Additionally, the method used for the

modernization will likely employ tools to help with the goal of automating many routine tasks when possible. Obeo, on their [Application Modernization Tools \(2019\)](#) web page, lists circumstances when tools are useful like automated migration, redesign, reverse specification, quality analysis, migration checklist, mapping and refactoring. These tools help to analyze and transform.

Harvesting Intellectual Capital

Many teams who perform application modernization work at the method-level (such as changing the UI of the application or reworking it for DevOps) invent tools and procedures. Once they are invented, they're often refined with many different customers over many projects and years. These practitioners create tools because they are close to what's needed, so they fill the gap. It can be both fun and effective.

How do organizations identify and gather these artifacts for use? The simplest way is to establish a knowledge network where members of the community can self identify and contribute materials. This is the minimum requirement.

A Better Knowledge Network

When a bunch of unedited contributions are made to a network, you quickly see that a more balanced solution is needed. For a more useful knowledge network, a knowledge engineer should be hired to perform specialized tasks, for example:

- Participate in the implementation, rollout and adoption of an innovative knowledge management solution
- Drive availability and adoption of knowledge from associates for use with customers
- Collect, synthesize and publish knowledge content onto the knowledge management system for use by the community
- Assist in automation of knowledge management to ensure knowledge is always up to date

You can see from this short list of examples why a knowledge network leader is desired. Their job is to maximize the usefulness of the material collected and reject material that doesn't meet minimum standards. Knowledge network leaders can usually connect a junior contributor with a mentor to get the work to an acceptable level.

Toolsets via Product Features

Some software and services companies advertise their toolsets for modernization as features of their products. Software AG does this with Adabas and Natural indicating that they provide a rich feature set to simplify application modernization, for example:

- Green-screen application modernization tools let you automatically transform green screens into modern Web pages
- Mainframe data connectivity enables you to access mainframe databases and non-relational data systems

These are just two of eight examples [World-Class Capabilities \(nd\)](#) web page.

Modern Systems takes a similar product approach to the notion of toolsets. An example is their [Mainframe COBOL Refactoring \(nd\)](#) program that uses their COBOL to Universal product that aligns with their services.

Semantic Designs (SD) is a software and services company with a tool called DMS Software Reengineering Toolkit. A sample of their [Software Engineering Services \(2018\)](#) includes consulting on customer problem scoping and definition of alternative software engineering solutions, consulting on customer project definition involving SD's DMS Software Reengineering Toolkit and application of DMS to general or specific re-engineering of application software suites.

Beyond Strategies, Methods and Toolsets for Modernization

When you examine strategies, methods and toolsets and think of them as part of a system, it becomes easier to see how they are used together to help achieve a project or program for application modernization. An effort to develop a strategy is use to kick off the process.

Once the targets for modernization are identified then one or more methods are employed to carry out the project like converting files to a DBMS or developing non-disruptive access to programs and data using APIs and microservices. Doing both of these projects would involve different methods. For every phase of modernization activity, a toolset should apply as there is a role to be played for tools from strategy work through DevOps deployment.

Strategies, methods and tools are a way to change and enhance applications to meet contemporary wants and needs. IT departments must not miss this organized and risk-managed approach to embrace the future.

References

7 Options to Modernize Legacy Systems. (2018). Accessed from <https://www.gartner.com/smarterwithgartner/7-options-to-modernize-legacy-systems/>

Application Modernization Services. (2019). Accessed from <https://www.360quadrants.com/software/application-modernization-services>

Application Modernization Tools. (2019). Accessed from <https://www.obeo.fr/en/services/modernization/346-application-modernization-tools>

Application Services. (nd). Accessed from <https://www.ibm.com/services/applications>

Legacy System Modernization: How to Transform the Enterprise for Digital Future. (nd). Accessed from <https://www.altexsoft.com/whitepapers/legacy-system-modernization-how-to-transform-the-enterprise-for-digital-future/>

Mainframe COBOL Refactoring. (nd). Accessed from <https://modernsystems.com/automatic-cobol-to-java-conversion/>

Modernization: Clearing a Path to Success. (2010). Accessed from https://www.standishgroup.com/sample_research_files/Modernization.pdf

Gulla, J. (2019). Modernization Is An Ongoing Strategic Activity. Accessed from <http://destinationz.org/Mainframe-Solution/Trends/Modernization-Is-an-Ongoing-Strategic-Activity>

Modernizing Transaction Banking. (2017). Accessed from <https://www2.deloitte.com/us/en/pages/financial-services/articles/modernizing-transaction-banking-technology-service-externalization.html>

Rationalize the Applications You Have. (nd). Accessed from <https://www.ibm.com/services/applications/modernization>

Semantic Designs. (2018). Accessed from <http://www.semdesigns.com/Products/Services/SoftwareModernization.html>

Software Engineering Services. (2018). Accessed from <http://www.semdesigns.com/Products/Services/>

World-Class capabilities. (nd). Accessed from https://www.softwareag.com/corporate/products/adabas_natural/appl_mod/default

About the Author

Joseph Gulla is an IT thought leader, researcher and writer. He publishes 50 BLOG posts and more than a half dozen articles every year for the IBM Systems Magazine and Destination z. Additionally, he works as a technical consultant on projects for clients in the US and from around the world.

Joseph is also the chief financial officer of Royal Swan Enterprises, a North Carolina Corporation that develops books, story frameworks, methods, products and services for young learners and their families. He also works as the general manager of Alazar Press, a publishing house that is an imprint of Royal Swan Enterprises. The press is dedicated to producing quality books for children of all ages and is aligned with the mission of Royal Swan Enterprises. As general manager, he is responsible for information technology, as well as contracts with authors, illustrators and partners. He also does the intellectual property work of the press.

Joseph has a MBA in MIS from La Salle University and a Ph.D. in CIS from Nova Southeastern University. He also has a certificate in Project Management from George Washington University. He has taught as an adjunct professor at the undergraduate, MBA and Ph.D. level since 1984. He also served on the Ph.D. committees of two students who successfully completed their dissertation projects.