

Business white paper

Applications Modernization

Adapting and embracing change to drive financial and service efficiencies



Table of contents

- 1 Getting started with assessments
- 1 Modernizing applications
- 3 Managing applications to increase value
- 3 Expected benefits
- 3 CASE STUDY:
Maryland Retirement
- 5 About the authors / More information

Secretaries of State encounter tight fiscal constraints annually, yet they must demonstrate leadership through improved service delivery by embracing new technologies or fall victim to increased costs, redundant systems, and inefficient services. Whether it is elections management, business services, agency administration, or the myriad of other responsibilities managed by Secretaries, opportunities for cost savings and efficiencies exist through modernization.

In a December 2011 Forrester Research study¹, more than 200 survey respondents identified three top drivers for modernization: cost; functionality; and the need for innovation. This research shows that agencies can improve service delivery and reduce costs by proactively evaluating their information technology infrastructure.

Getting started with assessments

Secretaries can begin by inventorying applications and infrastructure in place today to determine the value of each asset to the agency's mission and clientele. Establishing a baseline of assets and their value is the first step towards achieving your goals for increased efficiencies and cost savings, as well as ensuring compliance with security, governance, and operational standards.

With a complete inventory, Secretaries can quickly determine the ongoing benefit associated with legacy applications and systems by reviewing costs for support and maintenance, availability of experienced staff to operate and maintain each application, redundancy of applications and systems, and impact to users and clientele—building the business case for potential modernization. With new technologies emerging at a rapid pace, replacing legacy applications with new commercial off-the-shelf (COTS) applications may add more costs and risk to managing the agency's services; therefore, modernizing legacy applications may serve as a lower cost, lower risk alternative.

Application assessments are used to gain an understanding of the complexity and to develop a roadmap and a business case, showing the return on investment for the modernization strategies chosen. Holistic assessments account for multiple perspectives of a system—business, financial, functional, and technical. With this assessment, the systems can be mapped to the Secretary's tactical and strategic objectives.

Modernizing applications

Modernizing begins with assessing the complete IT environment including: applications; data; and infrastructure. This assessment is foundational. With this and the definition of the desired future state, Secretaries can match business goals and priorities to the information collected, determining the strategic value of each application to the business. This assessment searches out redundancies in applications and data, and begins the process for identifying opportunities for modernization. The next step is to recommend an approach for reaching the desired future

¹ "Application Modernization: Procrastinate At Your Peril! A Comprehensive Survey on Modernization in the Age of Mobile and Cloud."
— Forrester Research, December 2011.

state with the development of a Modernization Roadmap. This roadmap contains the series of projects that will be executed to achieve the desired outcomes. It also serves as the foundation for the business case which is necessary for business approval and buy-in.

Modernization strategies are scalable to each agency's needs and multiple strategies may be used to maximize value and return on investment. The table below categorizes and defines available strategies:

Overview of Modernization Strategies

Modernization Strategy	Definition	What does this really mean?
Re-learn	A discovery process that captures the intellectual property investment made in legacy applications over many years to preserve and carry it forward through modernization.	<i>Prevents the loss of current functionality and speeds requirements definition</i> of the application. This is useful for managing application updates going forward and as a precedent to other strategies.
Re-factor	Code optimization to improve the run-time efficiency of an application. Analyze legacy mainframe applications, create baseline metrics for performance measurement, analyze code performance at runtime, identify inefficient code segments and perform code restructuring, and provide updated code.	Improve performance by <i>optimizing the application by removing functions that no longer provide value.</i>
Re-host	Migration of legacy applications to lower cost modern platforms (such as cloud) without significantly changing the business features and functions.	Lowering ongoing hardware costs, including avoiding capital costs with cloud technology.
Re-interface	Re-interface legacy applications by creating new screened and non-screened interfaces to leverage and extend the application features and value.	<i>Connect to mobile and web technologies, and gain access to data</i> from other applications without conducting an enterprise systems modernization effort.
Re-architect	Forward engineering applications to modern Application Architecture to enable true enterprise agility.	Move applications for legacy high cost systems (for example, mainframe, COBOL) to lower cost technologies.
Replace	Provides a COTS framework to replace existing legacy applications with standard enterprise applications, industry standard applications, or newly developed applications.	Develop a transition plan to replace the legacy solution with a newer, lower cost solution.
Retire	Decommission legacy applications from the applications portfolio. Assess the linkage of applications to be retired and then systematically manage the shutdown of applications so that the business is not interrupted.	Complete successful replacement with transition from old application to new system.

Interim strategies such as re-hosting an application or re-interfacing to provide a mobile portal may provide the highest return on investment. Each Secretary who chooses to modernize will select the strategies that best meets their mission and goals.

However, longer term strategies such as re-architecting (forward engineering to a new language and/or architecture) provide the longest lasting benefits.

Re-architecting can be performed with automated tools, or by a combination of automated tools and manual re-factoring to optimize the code and increase application efficiency.

Re-architecting allows the most flexibility—the customer decides what business rules are implemented; the language to use; who will maintain it; when and how upgrades will happen; and whether or not the system will be hosted, cloud enabled or in a traditional local data center.

Replacing existing systems with COTS tools is also an option. Agencies may give up some control over business rules, base architecture and upgrade schedule, but gain speed of implementation.

The expected outcome from assessment is transparency into applications, data, and infrastructure inventory. This information can be used to make fact-based decisions. The financial planning data is used to develop business cases and model ROI. The end result is a modernization roadmap.

Managing applications to increase value

Once modernization is complete, the old systems no longer in use should be decommissioned. With the removal of the old systems, Secretaries will continue to demonstrate efficiency by reducing ongoing support costs and can redirect staff resources to higher priority projects.

To extend the new system's lifecycle, Secretaries will need to execute plans to ensure system documentation continues to be updated, follow development standards, and stay in step with architecture component upgrades. Without an active system management process, the new system will follow a similar lifecycle of the application it replaced—adding more costs and requiring more resources to maintain the system.

Expected benefits

In 18 months after modernization is complete, Secretaries should expect a reduction in operating costs (data center, hardware, and so forth) between 30-60%, a reduction in maintenance costs near 50%, and a reduction in the cost of new functionality development between 30-50%. Additionally there are other benefits: moving to a modern platform and language will provide a more secure application, and knowledge of the application will be captured in the system documents, making it easier to develop training materials for the system.

CASE STUDY: --- Maryland Retirement

Maryland's State Retirement Agency (SRA) administers death, disability, and retirement benefits on behalf of more than 350,000 active and former state employees, teachers, state police, judges, law enforcement officers, correctional officers, and legislators. The state of Maryland is the primary sponsor of this multi-employer defined benefit system; more than 100 local eligible governmental agencies voluntarily participate in the system as well.

Maryland identified the need to modernize its 30-year old legacy pension system. Like many other states today, Maryland faces an ever-increasing demand on state resources due to budget constraints and a growing number of state employee retirees. Among the significant challenges the state had while they operated the legacy system was supporting approximately 40 different pension and retirement plans for 115 employer agencies with changing needs, using old technologies that lacked agility.

Business rules mining (re-learn) was performed for the modernization of SRA's Maryland Pension Administration System – Phase I. More than 750 rules were extracted from an undocumented COBOL application with 600,000 lines of code and recreated in .NET (re-architect). The rules-based approach significantly improved delivery, quality, and customer satisfaction.

Carefully selected critical business functions, processes and tasks, were developed using a rule-based methodology that included detailed design specifications and delivery into the state's business rules engine. The business functional components selected for this method were benefit estimates, final benefit calculations, annual benefit statements, and annual posting of interest and service credits.

Case Study benefits:

- New agile technology provides access to more competitive resources and lower cost application updates.
- Positioned for future technology platforms, including web, cloud, and mobile
- Lower total cost of ownership and increased longevity of system lifecycle
- Improved user experience; screens have simplified application and data access decreasing the new user learning curve

For this project, a standard set of specification documents were re-used for each of the rule-based development tasks:

1. Rules document – Contained all business rules, annotated with clarification of complex cases or subtle distinctions between rules. Rules were organized into packages that corresponded to key business concepts, so they were easy to find.
2. Rule tables – This document laid out the rule tables that needed to be built into the Business Rule Engine.
3. Object model – Tables showing all objects and attributes needed by the rules. The object model was organized into objects that paralleled the objects used in the .NET framework.
4. Rule workflows – Flowcharts defined the execution sequence between the individual rule packages in the rules document, entry points, and exit points for system completion and validation failure.
5. Input data mappings – All object fields required by the rules were mapped back to system inputs or database fields.
6. Output data mappings – All output fields required by the system tasks were mapped back to object fields whose values were calculated by the rules.

Through standardization of these documents and the processes supporting them, Maryland has complete system documentation and can manage updates and enhancements more effectively by understanding the impact of proposed changes.

About the authors

Michael Boyd

Michael Boyd is the Elections Leader for Hewlett-Packard State and Local Enterprise Services. He is responsible for charting the strategic direction and directing the delivery for elections IT services for HP. He has extensive experience delivering IT projects successfully for state governments and previously served as the Governor's Policy Director in Mississippi.

Karen Gee

Karen Gee is the Applications Strategy Portfolio Leader for Hewlett-Packard State and Local Enterprise Services. In this role, she is responsible for charting the strategic direction and directing the delivery for Applications Modernization services for HP. She has delivered IT projects successfully for state governments, holding a diverse number of roles including developer, analyst and project manager.

Murali Kyasa

Murali Kyasa is a Senior Program Manager with Hewlett-Packard State and Local Enterprise Services. He led the modernization project for the Maryland State Retirement Agency.

To learn more :

Please contact the author via:
Email: michael.boyd2@hp.com
Phone: 919.622.8288

or Visit our website:

www.hp.com/go/SLED

Sign up for updates
hp.com/go/getupdated



Share with colleagues



Rate this document

© Copyright 2012 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Trademark acknowledgements if needed.

4AA4-xxxxENW, Month 20XX

This is an HP Indigo digital print.

