



## DEVOPS IN THE FEDERAL SECTOR

Saving the Federal Government millions of dollars through continuous delivery of software-driven innovation

October 2014



[www.reisystems.com](http://www.reisystems.com)

# TABLE OF CONTENTS

---

- Introduction ..... 2
- REI’s DevOps Approach ..... 2
- Implementing DevOps in the Federal Sector ..... 4
- REI’s DevOps Solution..... 6
- Benefits of DevOps in the Federal Sector ..... 9
- Challenges of Implementing DevOps in the Federal Sector ..... 11
- Conclusions ..... 13
- About the Authors..... 14
- About REI Systems, Inc. (REI) ..... 14
- About REI Solution Architecture Team (RSAT) ..... 14

## Introduction

---

**DevOps**, one of the latest approaches to software development (Dev) and operations (Ops), attempts to increase efficiencies, raise quality, and save money by eliminating traditional barriers between Dev and Ops teams. The promise of DevOps asserts that a more symbiotic relationship between these teams reduces the number of issues occurring during development, deployment, and operations, resulting in faster delivery of applications/features, and reduced Operations & Maintenance (O&M) costs.

As the government seeks innovative ways to leverage technology and increase mission efficiency under constrained budgets, DevOps is gaining traction in the federal market. Some proponents of DevOps mistakenly equate it to merely using an automated toolset, rather than considering its many facets and the critical elements required for successful implementation. DevOps success in the federal government is achievable through collaborative planning, the development of a robust governance structure, and the technical skills of the people involved.

**REI Systems, Inc. (REI)** has applied a DevOps approach to deliver complex systems to multiple federal agencies, and has gleaned lessons learned that can save the government millions of dollars by enabling continuous delivery of software-driven innovation. We enable developers to write and test code, and deploy useable software on the same day using *a variety of toolsets*.

This white paper provides a basic overview of DevOps, describes some of the principles and processes for successful implementation in the federal sector, and provides an example of an REI DevOps solution. Through our experience and lessons learned, we offer some key benefits for implementing DevOps in the federal government.

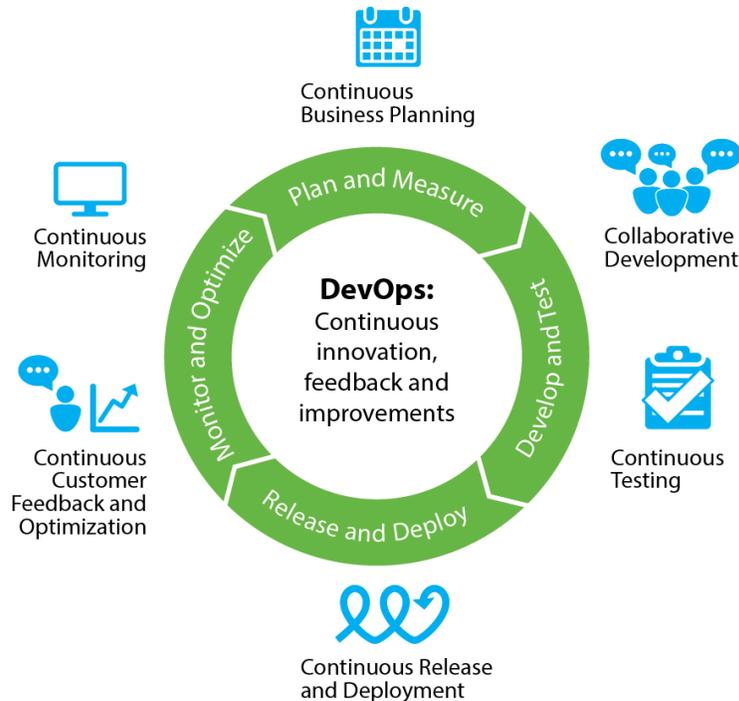
## REI's DevOps Approach

---

REI's DevOps solution uses a collaborative approach to minimize what can be a real or perceived division between development and operation teams. It promotes alignment of information technology (IT) activities to overall business objectives, and maximizes the ability to deliver more features while continually ensuring the stability of its systems. Our approach accelerates the ability of our customers to provide feedback on new capabilities under development, enabling the enterprise capability for continuous delivery of quality software. As a result, we expedite the delivery of critical information to improve mission accomplishment and decision-making. Continuous delivery (CD) is the ability to build, deploy, test, and release software products without manual intervention. CD supports DevOps by providing the

mechanism for continuous innovation, feedback, and improvements, while removing traditional barriers of separated teams, tools, and processes.

**Exhibit 1. The continuous innovation and continuous delivery lifecycle, enabled by DevOps**



Traditionally, Dev and Ops teams work in silos, in which Dev teams create the software in isolation and hand it off (deploy or release) to the Ops team to deploy it in a separate environment for their release. The Dev team is focused on creating software that is flexible enough to adapt to changing business requirements, whereas the Ops team strives to make sure the software is stable, reliable, available, and unchanging. Because the Dev and Ops teams have distinct objectives and somewhat dissimilar environments, they run the risk of duplicating effort, introducing inefficiencies, or uncovering new bugs. As a result, projects can suffer delays to resolve conflicts, manage scope creep, cost overruns, and quality issues. The following sections describe the processes that enable full-cycle continuous delivery in a DevOps solution as shown in **Exhibit 1**.

### **Collaborative Development and Continuous Testing**

Providing collaborative development and testing promotes the establishment of common business objectives for both Dev and Ops teams, so that both teams understand the impacts of their actions on each other and more importantly, the impacts on the success of the overall

business objectives. Integration efforts are reduced by continuously developing and testing the builds more frequently.

### **Continuous Release and Deployment**

Continuous release and deployment integrates the Dev and Ops processes to build, deploy, test, and release application code without manual intervention, which improves communication and coordination, further reducing both project risk and cost. Application acceptance testing can remain a manual process to ensure control over releases, however, unit testing, platform testing, staging, production, and post-deployment testing are automated processes. Increased frequency in deploying software to end users provides greater transparency and increased customer feedback to the end product resulting in higher overall customer satisfaction.

### **Continuous Monitoring and Customer Feedback**

Continuous monitoring and customer feedback improves customer satisfaction with the desired user experience, and provides the development teams with the information needed to develop the system with continuous feedback cycles of deployed code. Optimizing the applications with monitoring tools improves the availability and performance of the systems.

## **Implementing DevOps in the Federal Sector**

---

Applying DevOps to federal IT projects requires a thorough understanding of the factors that influence the federal IT landscape. REI's experience developing DevOps solutions has revealed several essential implementation considerations for federal agencies customers as they begin to formulate and execute a DevOps approach in their respective organizations.

### *Agency goals should drive toolset selection*

Agency goals and objectives should drive DevOps decision-making and toolset selection that provides for the appropriate level of flexibility for improvement and change over time. We have discovered across federal agencies that sometimes DevOps decision-making is mistakenly equated with a toolset that enables automation. While automation is a critical piece of DevOps, it is not the entire solution. DevOps encompasses a group of processes in the software development lifecycle (SDLC) each of which affects Dev and Ops environments differently. Agencies seeking a true DevOps solution will improve each piece appropriately to eventually reach a solution that enables continuous delivery. REI's DevOps approach connects processes and tools to enable fully tested, production-ready code that can seamlessly go live. REI's holistic, DevOps solution leverages the most appropriate tools to automate each phase of the

SDLC. As tools evolve and change, integration points change, as well, which often creates unanticipated difficulties.

While automation is a goal, REI also considers the cost of retiring or reusing existing investments, necessary skillsets required for change, consistent governance, and constant evaluation of new and innovative tools for improving performance and reducing costs.

### *Automate Common Tasks*

Code delivery, deployment, unit testing, security testing, and load testing should be employed in development, staging/integration, and production environments. Combining the teams creates an important synergy for automation; it sets the tone for successful collaboration because developers have the expertise to create the tools and operations has the expertise for running/maintaining the systems. If tasks are uncommon, but critical, automation may still be necessary. At minimum, teams should rapidly respond to these critical tasks by:

- Limiting the scope of changes to the system, reducing time to troubleshoot.
- Automating tasks, both common and uncommon, including the ability to auto test the production environment, and automatically revert changes upon test failure.

### *Use Rolling Subset Deployments*

Commercial companies like Google and Facebook use continuous delivery approaches to deploy updates to their core products as often as several times a week. To reduce risk, these innovative companies deploy new code to only a subset or sub-region of their user base. This rolling subset deployment allows for analysis of real customer reactions to the changes. Although many government applications do not have the user base of Google or Facebook, aspects of this philosophy can be applied in various application architectures to gauge the impact of changes and level of customer satisfaction.

### *Recognize differences between continuous delivery and continuous deployment*

Sometimes customers can become confused by the terms continuous delivery versus continuous deployment as they consider increased automation of the DevOps cycle. Continuous deployment, by definition, is to deploy every change to production immediately, whereas continuous delivery *means that every change is proven to be deployable at any time*. Often many hours have been spent verifying changes manually, going from Dev to staging for production. With DevOps, agencies have continuous visibility and continuous control regarding “go live” schedules and priorities. DevOps does not necessarily mean continuous change; in reality, a true DevOps solution will mean *continuous power to make the right changes*.

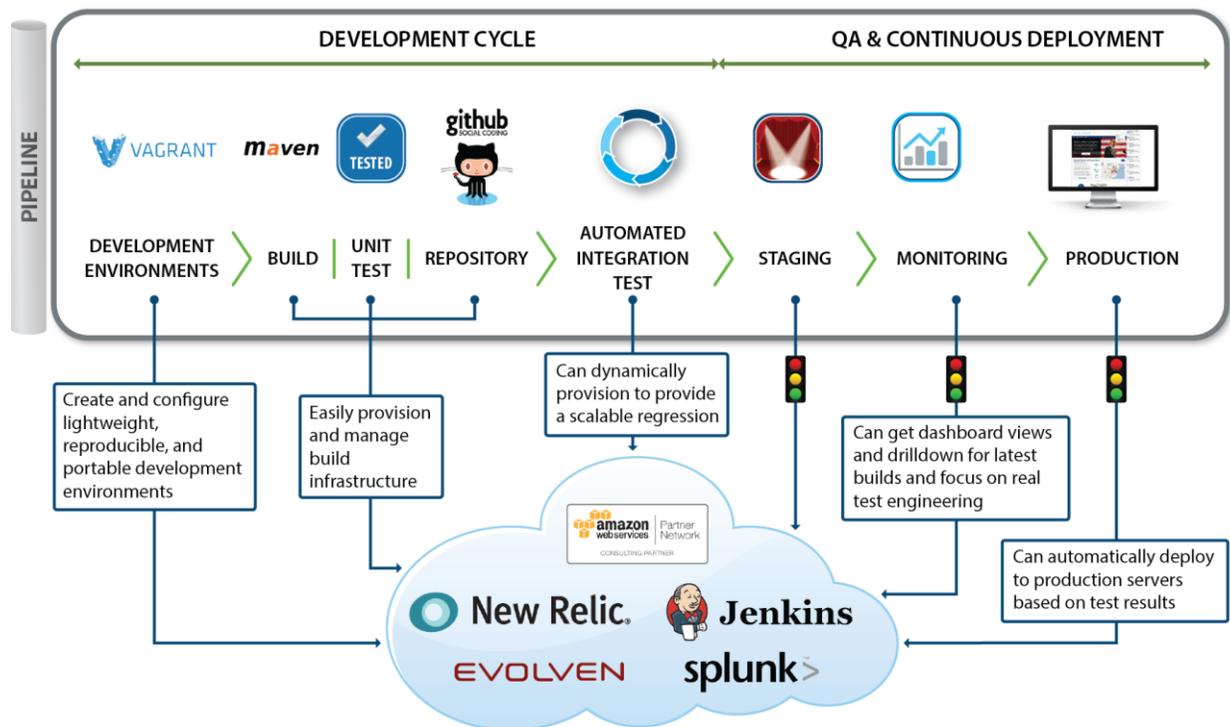
## Introduce Operational Transparency Innovations

For DevOps to work effectively, both Dev and Ops teams need to have the same visibility into systems. Logging analysis and presentation tools such as Splunk can be used to give developers real time, read-only access to system logs via a secure portal. In addition, DevOps teams can increase operational transparency; they are ideally configured to create metrics collection and analysis systems for both the application and the infrastructure, which frequently requires extensive customization to deliver the best results. Through operational transparency innovations, DevOps teams not only have the right monitoring tools in place, they also have the clarity to know *who to actually contact* to address problems and issues.

## REI's DevOps Solution

REI has applied a DevOps approach using a variety of tools for continuous delivery that include development environments, build, code repositories, automated testing, deployment, monitoring, and infrastructure automation. **Exhibit 2** features an REI DevOps solution with automated development and a continuous deployment cycle enabled on our DevOps platform. Using tools like Vagrant, REI quickly and easily stands up development environments. By establishing a common policy across all environments, no manual changes are necessary.

Exhibit 2. REI's DevOps Solution



Staging, User Acceptance Testing (UAT), and production are automated for faster deployment, and increased quality. Rather than using a single product, REI has leveraged a variety of tools to create an automated, continuous deployment solution. In the federal space, the use of open source tools eliminate the risks of purchasing monolithic tools and vendor lock-in. REI's DevOps platform reduces many licensing and maintenance fees, reduces development times and offers federal customers increased flexibility to integrate new tools as they become available, allowing agencies to change at the same pace as technology innovation. As needs evolve and priorities shift, agencies can switch out technologies for specific functionalities with little or no disruption to the development and deployment processes.

## **Development Environments**

REI's solution uses tools like Vagrant to instantiate Dev environments which allows us to provide a single, disposable, consistent environment which substantially increases efficiencies in integration. Using tools like Vagrant allows us to stand up Dev environments with easy-to-configure, reproducible, and portable Dev environments that can be provisioned on VirtualBox, VMWare, Amazon Web Services (AWS), or any other provider.

## **Build and Unit Test**

Developers need easy and repeatable ways of performing builds: we use tools such as Maven, and Ant, which provide the central mechanism for calling out to other parts of the delivery process, including dependencies, database changes, and static analysis, as well as creating environments and running tests. An integral piece of continuous delivery is the ability to perform continuous integration and testing. Providing unit tests for all software that is written enables developers to check in their code and ensures the software is tested against the baseline of code that other developers have been checking in and unit testing. Performing this continuous integration regularly exposes issues faster, and forces developers to communicate more effectively and understand the impact their changes have on the entire system. Continuous integration saves a significant amount of time resolving integration issues, and reduces risk by exposing and addressing issues regularly.

## **Repository**

As part of our DevOps approach, we strive for the most collaboration possible. Using tools like GitHub, we are able to provide powerful collaboration, code review, and code management for open source projects. These tools enable us to work together seamlessly to review code changes, comment on lines of code, and report issues. We can manage multiple teams on a

program; sharing code among team members reduces duplicate efforts and creates a common code base that can be shared across all team members and stakeholders.

## **Automated Integration and Testing**

Continuous Integration (CI) is a major component of our DevOps solution, and is our delivery pipeline which enables use for automated delivery of software at each stage, as represented by the stop lights icons in **Exhibit 2**, on page 6. REI can set up manual processes or automated delivery through each stage of the process. As errors are discovered, Dev and Ops teams are immediately notified, and can promptly rectify the errors. A component of our continuous delivery solution is automated testing. We use tools such as JUnit, Selenium, and Cucumber to perform unit, load, and performance, functional, and acceptance testing. We continuously apply other forms of testing including static testing for coding standards and security vulnerability testing to ensure availability, confidentiality, authentication, integrity, authorization, and non-repudiation.

## **Deployment**

Seamlessly and consistently deploying applications is a big part of the complete continuous development cycle. We use tools such as Capistrano to automate the process of stopping the servers, copying a WAR file, and configuring the application, and then restarting the server. Automating these steps enable continuous delivery, and pushes code as quickly as possible.

## **Monitoring**

REI offers a unique approach to monitoring using multiple tools to monitor the applications, environments, and systems. Using tools like Evolven and LMC for environment management provides real-time user tracking and work flows to manage the technical users that develop and operate the application on a daily basis. We use tools like Splunk for log analysis for developers and tools like New Relic to monitor the performance of the applications from the user's perspective, like page-load times, database-transactions, and systems monitoring to focus on CPU load, memory utilization, and disk space. Using Cloud environments like AWS allows our teams to better understand issues and metrics, and ensures that we are optimizing resources to reduce operational expenditures.

We can identify in real time the “when, what, why, and who” of change to enable efficient, Agile Management. REI's solution significantly reduces the enterprise risks and costs typically associated with continuous software releases of large, critical business applications. Our approach combines previously separate and often sequential shared processes across Dev and Ops organizations into a continuous, collaborative process.

## **Infrastructure Automation**

A key aspect to delivering DevOps is automating infrastructure tasks. Using tools like Chef and Puppet REI automates how to build, deploy, and manage the infrastructure. Using scripts to define the infrastructure, sets up repeatable, less error-prone processes.

## **Benefits of DevOps in the Federal Sector**

---

Through our experience applying the DevOps approach to federal IT projects, REI has demonstrated real business benefits that address common concerns for most federal IT departments including: resource constraints, quality and adoption issues, reduced time-to-market, program revenue increases, availability, and saving money. The following section describes some of the high-level benefits REI has realized for federal customers.

### **Benefit 1. DevOps puts federal agencies on the leading edge of technology innovation**

While some customers continue to struggle implementing Agile development methods, other agencies have taken their technology practices to a new level by augmenting Agile development with DevOps. While using Agile methods can speed up the development process, its collaborative approach focuses more on requirements gathering and development, and since Ops has typically not been an integrated part of the development process, deployment into production can be slow and flawed. Bringing DevOps capabilities into the enterprise eliminates this issue and allows agencies to reach a new level of performance leveraging the benefits of using Agile techniques and methods.

### **Benefit 2. DevOps reduces technical debt**

Technical debt can be defined as the cumulative costs of deficient software architecture and development code. Many federal agencies are beginning to feel the pressure of high accumulated technical debt as they attempt to adopt new technology. If current debts are not addressed, they will produce additional issues, making it difficult to rectify later. When technical debt is not addressed, software can become unpredictable and completely disordered. These cascading effects can impact future IT investments, as money that could have been used for future developments has to be repurposed to repair technical debts and keep existing systems functioning.

Technical debts contribute to a sense of apprehension in making future investments because agencies may feel burned by the problems they face now from legacy IT investments. REI's DevOps approach can help reduce technical debt for the government by designing systems that can adapt to volatility, rather than building systems to avoid anticipated stressors. We know that events that occur infrequently are introduced to the system continuously during development, therefore our philosophy is to empower development and operations teams to handle events early in the process, proactively shoring up system vulnerabilities in real time. This provides the basis for both the development and operations teams to improve on their ability to handle these events and improve response times. This practice occurs continuously throughout the SDLC. Our approach is based on countless lessons learned: systems should not only survive anticipated events and failures, but should be built to improve incrementally as they undergo continuous incremental change and automated testing of as many scenarios as teams can anticipate. As a result, systems can become increasingly robust with every update.

“REI's DevOps approach can help reduce technical debt for the government by designing systems to adapt to volatility, rather than build them to avoid unanticipated stressors.”

-Scott Fletcher  
Chief Operating Officer  
REI Systems

### **Benefit 3. DevOps decreases time-to-capability**

**Prevention** is the key word when thinking about how REI uses DevOps to reduce time to market. By combining a key goal of the Ops team, “to maintain a system’s reliability” with a key goal of the Dev team, “to develop new functions,” we focus on establishing automated procedures that prevent software glitches resulting from deploying defective code into production. REI's preemptive approach to software development ensures that both teams are working toward the same goal and are also considering the impacts of their actions on each other's work processes. We know this approach results in fewer problems that require reexamination and retesting and ensure that the code is developed and deployed the right way, the first time.

### **Benefit 4. DevOps saves money by building in quality early**

REI's DevOps approach incorporates quality from the onset of the development process. As mentioned previously, Dev teams aim to develop new functions while Ops teams try to maintain reliability and stability. By combining each team's focus at the beginning of the process, controls are established early on which ultimately prevents the delivery of defective code, saves money, and maintains trust among both Dev and Ops teams.

## Benefit 5. DevOps delivers better quality

DevOps at REI not only builds quality into the process earlier, it also promotes higher quality levels overall by:

- **Creating constancy of purpose.** REI's focus on continuous delivery forces Ops and Dev teams to implement proactive approaches to reduce vulnerabilities and to improve responsiveness to unforeseen events.
- **Ceasing dependence on inspection to achieve quality.** The traditional Dev approach has long intervals between deployments with major changes, requiring extensive regression testing. Our DevOps approach does not depend on lengthy inspections because we encourage continuous deployment of incremental changes.
- **Functionally combining the Dev and Ops teams despite separation by contract/vendor.** REI has worked closely with federal agencies to functionally combine teams through a DevOps approach on separate contracts to achieve a common goal.
- **Continuously refining and improving every process.** Constant improvement of every process is fundamental to REI's DevOps approach. Our approach encourages our DevOps teams to consistently learn on the job by responding quickly to even minor events, while simultaneously reducing the likelihood of major outages and stretching their holistic understanding of the entire system.
- **Eliminating the blame game and fear of deployment.** Our DevOps approach focuses on continuous delivery and shared accountability, and eliminates the fear of application deployment because it becomes a routine occurrence.

“DevOps really is key with the pace of today's development environment. By integrating teams and quality early in the process, providing better communications, purpose and direction, continually refining processes, it greatly improves outcomes and stops the blame game.”

-Keith Thurston  
General Services Administration (GSA)

## Challenges of Implementing DevOps in the Federal Sector

---

While DevOps can bring quantifiable benefits to the government, implementing it within the federal sector requires an understanding of their unique challenges.

### Challenge 1. Risk Averse Paradigm

Federal government IT decision-makers are hesitant to introduce new approaches to technology because they are bound by constraints and responsibilities that their private sector counterparts do not need to consider. Accountable for taxpayer dollars, the government has a

significantly larger group of stakeholders with varying priorities. As a result, the federal government may resist the adoption of DevOps until they rigorously vet the approach. For DevOps to gain traction in the federal sector, its value must be demonstrable and quantifiable.

### **HOW REI ADDRESSES THIS CHALLENGE**

REI helps agency decision-makers demonstrate the efficacy of DevOps through the use of metrics. We use number and frequency of software releases, volume of defects, time/cost per release, number and frequency of outages and performance issues, the mean time to repair, or to resolve issues, and cost of resources to baseline an agency's existing position. We then work with the government to establish performance metrics goals, and monitor progress toward these goals by comparing the baseline to new metrics at established intervals. As a result, agencies have a clear picture of how DevOps improves quality and increases efficiencies.

### **Challenge 2. Governance**

Adoption of DevOps is a major opportunity for the government to efficiently deliver high-quality software; however changes in software development approaches also signals required changes in governance structures. The governance processes used by organizations to define and manage their software development lifecycles are very different across agencies. To bring DevOps into the federal sector, agencies must implement changes to their integrated Governance, Risk, and Compliance (GRC) frameworks.

### **HOW REI ADDRESSES THIS CHALLENGE**

For DevOps to be successful, a GRC framework must be designed to facilitate its implementation. REI leverages lessons learned to help agencies define and manage the DevOps paradigm, evaluating business and regulatory risks and controls and monitors mitigation actions from the DevOps paradigm. This ensures that the processes and internal controls are in place to meet the requirements imposed by governmental bodies, regulators, industry mandates, or internal policies. We have successfully designed GRC frameworks to support a DevOps approach for federal agencies.

### **Challenge 3. Federal Contract and Incentive Structures**

Federal contract structures can inhibit the integration of teams because incentive structures are defined differently and separately for Dev and Ops teams. Often, Dev and Ops teams are comprised from different vendors with no vested interest in the success of the other team.

## **HOW REI ADDRESSES THIS CHALLENGE**

REI works with agencies to create an incentive structure that promotes mutual accountability. While separate groups will still have different access privileges, the Ops team will be required to give the Dev team real-time access to see how the deployment is playing out. By creating the appropriate incentive structure, agencies can still maintain isolation of duties while enabling real-time visibility.

## **Challenge 4. Immediate Vs. Long Term Priorities**

Federal agencies face constant challenges of competing priorities--between choosing to focus on immediate/functional requirements or on investing in long-term solutions. With some IT solutions carrying large technical debts, the decision to recover legacy projects or invest in something new introduces heavy considerations. While DevOps reduces technical debts and brings long-term benefits, its introduction to government may not be an immediate priority.

## **HOW REI ADDRESSES THIS CHALLENGE**

REI demonstrates how the implementation of a DevOps approach actually simplifies the prioritization of IT projects. Because our DevOps approach is not limited by a toolset, REI alleviates the burden of considering a large, new investment and helps agencies automate the parts of the SDLC that they require and can afford at the time. REI's DevOps solution delivery pace is flexible enough to match the pace desired by our customers.

## **Conclusions**

---

Implementing DevOps in federal agencies requires a vendor that understands government challenges, and can build a flexible solution using multiple toolsets to meet these challenges. REI's focus on the elements that truly represent a holistic DevOps solution: toolsets, skills, and governance, ensures that our DevOps solution elevates our customer agencies to the leading edge of technology innovation. REI's approach allows federal agencies to leverage technology in new ways to gain added value from investments, gain efficiencies through improved processes, and provide improved transparency into system development processes that can substantially improve quality, and reduce project risks and costs.

**For more information on REI's DevOps solutions, contact David Callner, [david.callner@reisystems.com](mailto:david.callner@reisystems.com), (703) 574-9273.**

## About the Authors

---



David A. Callner delivers focus, strategic excellence, and leadership expertise acquired over a distinguished 15+ year career in Information Technology. Mr. Callner is REI System's Chief Architect and is accountable for developing solutions, technology strategy and roadmaps for large government programs. Mr. Callner received a M.S. in Computer Science from Johns Hopkins University, and his B.S. in Computer Science from University of Texas.



Dr. Shanthy Edward is a content strategy and organizational development expert with over 10 years of experience developing white papers, process plans, organizational development strategies, training curricula, marketing materials, and proposals. Dr. Edward received her Psy.D. and M.A. in Clinical Psychology from the Illinois School of Professional Psychology and her B.A. in English/Sociology at the University of Western Ontario. She is ITILv3 certified.



Mr. Michael Iams is a solutions architect with over 17 years of experience in designing, integrating and supporting Internet based web applications. Mr. Iams is an expert in Internet and Cloud infrastructure system design as well as IT Security. Over the last ten years, he has designed and operated infrastructure for over two dozen federal websites, including Data.gov, SBA.gov and USASpending.gov. He received his B.S. in Finance from the American University.

## About REI Systems, Inc. (REI)

---

REI is a leading provider of web-enabled business solutions for the federal government and the commercial sector. Drawing upon our mission to deliver reliable, effective, and innovative solutions, the company works daily to solve complex homeland security, defense, health resources, criminal justice, and performance support issues. REI is strengthened by its commitment to exceeding customer expectations. REI has been serving government at all levels and major corporations since 1989. For more information, visit [www.reisystems.com](http://www.reisystems.com).

## About REI Solution Architecture Team (RSAT)

---

RSAT is a group of skilled technical and business professionals who focus on generating new technologies, developing innovative solution approaches, and proactively addressing unmet customer needs in relation to high-priority government IT challenges. Contact RSAT at [solutions@reisystems.com](mailto:solutions@reisystems.com).