The road to agility
Getting to full automation with enterprise applications
01 Agility: The ultimate destination
02 Enterprise automation: The road less travelled
03 The map: Start with the destination in mind
04 The automobile: A model for automation
   Lessons from the assembly line
   Road-tested results
05 The journey never ends
06 How we can help
As consumers, we benefit from the agility of modern consumer-based applications virtually every day. E-commerce sites scale seamlessly to handle massive spikes in holiday shopping traffic. We tune in to stream TV releases viewed by millions of households, possibly consuming half the bandwidth on the internet. Popular social media sites release thousands of small updates daily with software tweaks tailored to specific user populations.

The agility, velocity, and scalability of these consumer services are supported by modern engineering practices that are, at their core, based on the mantra of “automate everything.” The engineering practices underlying these capabilities are well documented and, in many cases, supported by robust open-source software that can be used to achieve significant levels of automation in any organization, including large enterprises. Why then, do many enterprises fail to deliver the same level of agility, velocity, and scalability to their applications’ users?
As it turns out, the wealth of automation knowledge, capabilities, and software is not equally distributed. The smaller, more modular, and often open-sourced software frequently preferred by well-known, consumer-focused companies was built with automation in mind, or it has been integrated with open-source automation tools. The software used by large enterprises is often commercial off-the-shelf (COTS) software that is packaged up as a composite business application using other software components. While this enterprise software is feature rich and business focused, it was typically not built with automation in mind.

Enterprises aspiring to higher levels of automation with traditional COTS software or hybrid COTS / open-source solutions typically encounter one of three scenarios:

1. **Automation prohibitive**
   Some or all of the software components of an enterprise application are either proprietary, significantly complex to access, or cost prohibitive to automate. For this software, traditional release and deployment processes continue to be employed.

2. **Automatable, with some effort**
   This software often has open configuration mechanisms and programming interfaces, but they are not built with automation in mind and may be poorly documented for these purposes.

3. **Automation native**
   This is software that was built with automation in mind or was retrofitted with APIs and other automation interfaces with the explicit purpose of supporting a high degree of automated configuration and deployment.
The map
Start with the destination in mind

As we embarked upon the creation of a software-as-a-service (SaaS) offering built on IBM® Cloud and IBM enterprise software, we did so cognizant of our customers’ expectations around rapid and agile software delivery. To measure our performance, we set a goal of 12 hours for complete readiness of a client instance of our solution—including cloud provisioning, complete software installation and configuration, and deployment of the solution, dashboards, and APIs. (See the case study, “Road-tested results,” for more information.)

The combination of “automatable, with some effort” and “automation native” products in our solution meant that a fully automated provisioning cycle within 12 hours was achievable, but not without a plan, quality design and engineering work, and support from IBM. The engineering team crafted the automation solution using the metaphor of automobile engineering and assembly: specifying a model to build, defining a platform or “chassis,” implementing standard automation, and then augmenting with custom automation where needed.
We set a goal of 12 hours for complete readiness of a client instance of our solution—including cloud provisioning, complete software installation and configuration, and deployment of the solution, dashboards, and APIs.

The map
Automation reduces provisioning times
Our team embraced the idea of describing its software automation efforts in terms usually reserved for the factory floor. The team defined four steps for constructing the automation model for our solution:

1. Specify the model
Automobile factories are dedicated to building one or two car models at a time, plus variations (e.g. base, luxury, sport). One of the first things our team did was specify the quality attributes of our product and define the families of product variations that could be customizable by clients. As an example, solution portability was one of the most important quality attributes, which led to a decision to use VMWare® ESX and NSX in the IBM Cloud or an on-premise configuration, based on customer requirements.

2. Design a reusable platform
Akin to automobile chassis that are used as a common platform for several brands and models of automobiles, we identified a common platform that we would use as the foundation to plug in the individual components of our solution. The market for automation software to support our platform buildout is extensive; researching all options would be time intensive. Instead, our engineering team identified a core automation platform, invested heavily in that platform, and augmented as necessary with automation scripts and hooks. Aligning with our model decision and experience, we elected to use vRealize Automation to orchestrate both IBM Cloud’s public cloud infrastructure and IBM Software provisioning and configuration.

We set out to build an enterprise automation framework that enables fast deployment, rapid releases, and agility within a complex enterprise commercial off-the-shelf technology (COTS) environment.
The automobile
A model for automation

3. Employ standard parts
Our team viewed our solution’s software components as “automobile parts” and looked for ways to optimize their alignment with our solution and automation platform. The team built an automation blueprint and then plugged in the components using available automation and configuration interfaces. The team’s goal was to maximize the number of standard parts and automate these parts. These standard parts aligned with the “automation native” capabilities.

4. Craft your own parts, where needed
The secret sauce of our automation—and, we believe, of any enterprise automation—is addressing the components that are “automatable, with some effort.” At this point in the process, our team had built an 80 percent complete assembly line and needed to “machine” the final parts of our automation solution. The remaining 20 percent of the automation required significantly more effort to achieve and resulted in a number of lessons learned.

Our team viewed our solution’s software components as “automobile parts” and looked for ways to optimize their alignment with our solution and automation platform.
The road to agility

Lessons from the assembly line

While customizing and tailoring our platform automation, several valuable lessons surfaced.

1. Nurture supplier relationships

Our close working relationships with leading technology companies, like IBM and VMware, enabled us to call their technical experts who could explain the inner workings of their products. They helped us figure out how to automate things that were not intended to be automated and readily accepted feedback on how to improve their products to make them more amenable to future automation.

2. Adopt a DevOps culture

DevOps and automation engineering are mindsets that need to come from within the team. Similar to the Japanese Kaizen movement that resulted in continuous improvements to Toyota’s manufacturing process, DevOps is a culture that permeates from management through the entire team.

3. Prepare for next year’s model

Our team built the automation process using tools that were established during project initiation. With the speed of change in technology, we should always look ahead and incorporate new capabilities as they become viable. As a result, we see containerization and cloud-native capabilities on the horizon for our next “model.”

4. Align the “workforce”

Although it’s hard to think of our engineers as a workforce in a traditional sense, there is a need to make sure that we think about aligning their skills and capabilities with our automation approach. The same goes for preparing for next year’s model; new technologies require training and additional skills.
In 2016, the Centers for Medicare and Medicaid Services (CMS) reinforced their modularity requirements for new Medicaid Enterprise Systems (MES). This led to the emergence of a number of smaller, more focused MES modules that could be integrated into a composite Medicaid application.

**The challenge**
In addition to modularity, CMS enforces a number of other standards and conditions for its systems, including a more aggressive security posture and the use of proven COTS products. While this makes for more repeatable, business-focused solutions, these products often fall into the “automatable, with some effort” category, which is not unusual for regulated industries, including healthcare and financial services.

**Our response**
Deloitte understands the dynamics of operating in a regulated industry and is committed to CMS’s standards, including modularity. Our modular MES solution was built with industry-standard COTS products on a cloud platform. We worked through the “automatable, with some effort” software to produce a modular solution for states that could be deployed in a manner of hours, not days or weeks.

**The results**
Our MES solution can be delivered in a rapid and agile environment and configured to meet the specific needs of individual states. It aligns with CMS standards and conditions and is built to support service-based integration of other modules, enabling states to dynamically change their solution configuration. The solution is being rolled out across the country and is being continually evaluated for enhancements and improvements to better meet client needs.
The journey never ends

A year into our automation journey, with our goal of a complete platform deployment in 12 hours in the rearview mirror, we continue to explore ways to improve the process. We are amazed at the speed at which IBM and the rest of the industry has rushed to support containerization. Cloud adoption continues to increase, including in regulated industries and government. At the same time, we recognize there will always be a place for highly skilled engineers to provide innovative and resourceful solutions to address the automation of complex enterprise software deployments.
How we can help

How the Deloitte and IBM alliance can help
For almost 20 years, Deloitte and IBM have helped global enterprises address their toughest business issues. This “smarter teaming” approach often results in better service and higher value for our joint clients. Our unique alliance unites the depth and breadth of IBM’s technology portfolio with Deloitte’s practical, innovative solutions. By working together, Deloitte and IBM offer customized solutions focused on the business issues important to our clients.

If you’d like to learn how to make your enterprise IT environment more agile—please contact us.

Authors
Thomas Beck
Principal
Public Sector
Deloitte Consulting LLP
thbeck@deloitte.com

Van B. Spaulding
Senior Manager
Senior Solution Architect
Deloitte Consulting LLP
vspaulding@deloitte.com

Jesse Holbrook
Manager
Solution Architect
Deloitte Consulting LLP
jholbrook@deloitte.com

Contributors
Dave Knight
Senior Solution Architect
IBM Alliance
Deloitte Consulting LLP
daknight@deloitte.com

Brooke W. McCloskey
Senior Manager
IBM Alliance
Deloitte Consulting LLP
bwmccloskey@deloitte.com
About Deloitte
As used in this document, “Deloitte” means Deloitte & Touche LLP, a subsidiary of Deloitte LLP, which provides audit and risk advisory services. Please see www.deloitte.com/us/about for a detailed description of our legal structure. Certain services may not be available to attest clients under the rules and regulations of public accounting.

This document contains general information only and Deloitte is not, by means of this document, rendering accounting, business, financial, investment, legal, tax, or other professional advice or services. This document is not a substitute for such professional advice or services, nor should it be used as a basis for any decision or action that may affect your business. Before making any decision or taking any action that may affect your business, you should consult a qualified professional advisor. Deloitte shall not be responsible for any loss sustained by any person who relies on this document.

Copyright © 2018 Deloitte Development LLC. All rights reserved.