



WHITE PAPER • MARCH 2018



CA Continuous Delivery Automation at Scale

How to Choose a Solution That Grows With Your Needs

Table of Contents

Executive Summary	3
Section 1	4
Administrative Scalability	
Section 2	4
Functional Scalability	
Open Plugin Architectures	
Verified Community Contribution	
Section 3	6
Geographic Scalability	
Scalable User Sessions	
Deployment Automation Targets	
Payload Delivery	
Geographic Queued Deployments	
Geographic Data Access	
Section 4	7
Load Scalability	
Section 5	8
Generational Scalability	
Section 6	9
Conclusion	

Executive Summary

CA Continuous Delivery Automation tools enable best practices in deploying and promoting application-related artifacts, properties, configurations and even data across an application lifecycle in a safe, predictable and repeatable manner. Gartner reports that using an CA Continuous Delivery Automation tool is key to enabling DevOps and achieving continuous delivery. The challenge enterprises face is that they fail to consider the scalability of the CA Continuous Delivery Automation tool they purchase. Within a couple of years of implementing the solution, many companies reach the scalable limitations of the product they selected and find themselves abruptly at the end of their continuous delivery runway, looking at creative ways to “hack” their way around the limitation, or even buying another CA Continuous Delivery Automation tool from another vendor.

In other words, underestimating the importance of scalability for CA Continuous Delivery Automation tools costs money and time. To help you avoid selecting an CA Continuous Delivery Automation solution that will not scale to both your immediate and future needs, we will be discussing what “scalable” means in an CA Continuous Delivery Automation context. Wikipedia defines scalability as “the capability of a system, network or process to handle a growing amount of work or its potential to be enlarged to accommodate that growth.” Wikipedia also defines five dimensions of scalability that can apply to

CA Continuous Delivery Automation tools:

- Administrative scalability
- Functional scalability
- Geographic scalability
- Load scalability
- Generational scalability

This white paper will explain how each of these criteria is relevant when choosing an CA Continuous Delivery Automation product that is capable of scaling along with the growth of your enterprise.

SECTION 1

Administrative Scalability

Wikipedia defines this dimension as being able to increase the number of organizations/users on a single distributed system. In the context of enterprise software, this relates to the overhead of administering and maintaining the solution itself. At the enterprise scale, one must always consider whether the CA Continuous Delivery Automation tool of choice is not only capable of being used by other business units but by a diverse set of technical and non-technical specialists who are involved in releasing software into production.

The product must be intuitive enough for non-technical personnel to view and report on past/present deployments and the status of current releases, as well as to simply be able to approve a release and request a deployment. An CA Continuous Delivery Automation tool must provide self-help capabilities for non-technical users while also allowing technical users the ability to drill down into automation details for troubleshooting forensics. Additionally, the most technical of all users, developers, need command line interface (CLI) and application programming interface (API) access to the CA Continuous Delivery Automation tool so it can seamlessly integrate into their software development lifecycle (SDLC) tooling and practices.

The ability to scale to a diverse set of users is important for onboarding additional BUs throughout the enterprise. This brings up the next item for consideration: Can the CA Continuous Delivery Automation tool you select easily handle a growing number of distributed users, teams and departments? Does the CA Continuous Delivery Automation tool support multi-tenancy, so separate business units, departments or teams can remain independent of one another?

Multi-tenancy is the most underrated feature of an CA Continuous Delivery Automation solution. It allows:

- Scaling out usage while maintaining secure separation if required
- The flexibility to place production deployments in a separate tenant from non-production deployments to provide better security for, and isolation of, production environments
- The same automation mechanics to exist in all environments—a core concept of continuous delivery—while also providing the enterprise with greater security and safety

Last, the scalability of user management must be considered. It can become laborious and disjointed if your CA Continuous Delivery Automation tool does not seamlessly integrate with your active directory (AD)/lightweight directory access protocol (LDAP) services. Without AD/LDAP integration, IT personnel has to regularly maintain multiple credential and role-based access control (RBAC) stores. This causes a delay for adding new users and disrupts work in progress for IT staff. When considering the administrative scalability of an CA Continuous Delivery Automation tool, AD/LDAP integration is a must. The whole point of buying an automation solution is to reduce labor-intensive or interruption-driven work. You do not want to trade administrative overhead for process efficiency.

SECTION 2

Functional Scalability

In the context of CA Continuous Delivery Automation, this means the ability to extend your CA Continuous Delivery Automation tool to integrate, automate or orchestrate the DevOps toolchain with minimal effort. That is, once you have the tool installed, you will want to include many, if not all, of the supportive tools in your app's ecosystem, whether it is a version control system (VCS), a hypervisor, an infrastructure-as-a-service (IaaS) provider or a database. Selecting an CA Continuous Delivery Automation tool that easily allows your organization to quickly onboard more technologies into a coherent pipeline is critical. Several CA Continuous Delivery Automation tools are designed with the developer/development use case in mind, rather than considering a broader enterprise IT context. These

tools have short runways, and companies quickly become dissatisfied as their CA Continuous Delivery Automation needs outgrow the usefulness of the CA Continuous Delivery Automation solution they purchased.

There are four key requirements that enterprise CA Continuous Delivery Automation tools should have:

- An open plugin architecture
- A mechanism for verified community contributed plugins/extensions
- Extensibility outside of plugins
- Ready-made, best-practice workflows or execution paths

Open Plugin Architectures

Plugins are all the rage for CA Continuous Delivery Automation tools. Plugins reduce the amount of scripting and expertise required to stitch together a deployment process. The challenge is that the plugin architecture provided by many CA Continuous Delivery Automation vendors does not allow you to see the code inside the plugins. What if you decide to switch CA Continuous Delivery Automation providers? What if you need to perform what the plugin does manually? Enterprise IT needs this kind of visibility and flexibility. Plugin architectures that are a black box are no good and vendor-lock your organization. Without visibility into a plugin's execution, it may be impossible to understand why a deployment task failed in the worst-case scenario, and it increases the mean time to resolution (MTTR) in a best-case scenario.

Additionally, visibility into the automation mechanics of CA Continuous Delivery Automation plugins provides valuable examples of best practice integration procedures. Every IT professional is always looking to follow industry best practices, and openly architected plugins are a great way to gain understanding and learn from subject matter experts. Also, you will reach a point when a plugin doesn't exist for the tech stack you want to include in your deployment pipeline. Existing plugins provide a great learning library from which to begin writing your plugin.

Last, most CA Continuous Delivery Automation tools require you to learn yet another programming language. Many of the languages used by vendors are popular now, but what about five to 10 years from now? The long-term viability of languages such as Groovy, Ruby, Perl and other open-source languages is not proven, and their popularity wanes or fluctuates like fashion fads (see TIOBE or PYPL Indexes). When it comes to developing an enterprise-class continuous delivery pipeline, you must consider the language constraints the vendor imposes upon you to create plugins or augment the ones supplied out of the box.

Requiring your staff to learn another language doesn't scale well because it increases the qualification requirements for finding and hiring new employees. As cool as plugin languages are, no matter how popular they may be today, it is just another thing DevOps personnel need to master. Five years from now, you may find it hard to hire talent with the requisite programming skills to maintain your CA Continuous Delivery Automation plugins. A scalable CA Continuous Delivery Automation plugin architecture will provide a thin layer above time-proven CLIs. CLIs require little ramp time to learn, have been around for decades and are almost always provided by the middleware or tech stack you want your CA Continuous Delivery Automation tool to automate or orchestrate.

Verified Community Contribution

Selecting an CA Continuous Delivery Automation tool from a vendor that allows its customer base to contribute to the plugin marketplace is paramount. No one vendor or customer can produce all the integration code for the growing DevOps toolchain. It takes a community of problem solvers who wish to help their fellow IT professionals solve the same problem they just solved. One of the promises of CA Continuous Delivery Automation is to avoid reinventing the wheel over and over. Surely someone at some company somewhere has come across the technology or situation that you face, right? Community contribution helps the functional scalability of your CA Continuous Delivery Automation solution.

However, community-contributed plugins must be warranted, verified and reviewed by the CA Continuous Delivery Automation vendor, or disaster can strike. You don't want to install a plugin that is a Trojan horse, exposes a backdoor or is part of a sophisticated hack. In the modern era we operate in, plugins not provided by an CA Continuous Delivery Automation vendor must be checked and warranted safe by the CA Continuous Delivery Automation vendor. If an CA Continuous Delivery Automation vendor does not give any level of warranty for community plugins, it is safer to develop the plugins yourself. No organization can afford to assume the risk, because CA Continuous Delivery Automation tools touch every environment within IT, including production. The exposure is too high and the risk is terrible.

The potential harm of community plugins underscores the earlier point about the importance of multi-tenancy in an CA Continuous Delivery Automation tool. A multi-tenant CA Continuous Delivery Automation solution will allow you to test plugins in an isolated tenant without exposing your live continuous delivery pipeline to risk

SECTION 3

Geographic Scalability

Scalable User Sessions

Your CA Continuous Delivery Automation tool must be able to accommodate your distributed workforce. Designing release pipelines requires one or more users to log in and design a release into succinct phases or stages and create tasks, activities and milestones. An ideal CA Continuous Delivery Automation tool will allow you to assign users to said tasks/activities, define approval gates and provide progress reporting. Release management work requires longer user sessions than deployment automation activities. The release management capabilities of an CA Continuous Delivery Automation tool need to be able to maintain responsiveness no matter where users are geographically located and should scale to meet the number of simultaneous users logged in. An CA Continuous Delivery Automation tool's user session state should be able to work easily from behind load balancers or regional DNS servers. In theory, modern web-based UIs provided by CA Continuous Delivery Automation tools should have no problem scaling user sessions. That said, you should confirm the CA Continuous Delivery Automation tool you select meets your user demographic needs or expectations.

Support for ChatOps can reduce the number of users that must regularly and directly log into an CA Continuous Delivery Automation tool. An enterprise CA Continuous Delivery Automation tool that provides ChatOps must allow an IM user to check the status of releases and automation executions, approve deployments or activate any self-help features the CA Continuous Delivery Automation tool offers. Logging into a system and performing multi-click drill-downs to perform repetitive and straightforward CA Continuous Delivery Automation tasks is much faster from a chat client like Slack or HipChat. It's not that a lack of ChatOps support needs to be a deal-killer when selecting an CA Continuous Delivery Automation vendor, just that ChatOps is a geographic scalability factor for CA Continuous Delivery Automation. If your organization uses ChatOps or is planning on using it, then you ought to select an CA Continuous Delivery Automation tool that supports ChatOps.

Deployment Automation Targets

All the hype surrounding containers and microservices is reminiscent of the hype during the late 1990s concerning the end of mainframes. If technology history teaches us anything, it's that the enterprise IT portfolio of the future will be even more disparate, diverse and distributed than it is today. Containers and microservices will not replace anything; they will just be an additional member of the portfolio. CA Continuous Delivery Automation tools must be able to deploy to a cornucopia of endpoints hosted anywhere while maintaining process throughput and responsiveness. Enterprise CA Continuous Delivery Automation tools need to be like a powerful vacuum that doesn't lose suction when clogged.

Small companies, simple release use cases and cloud-only apps may be able to get by with an agentless CA Continuous Delivery Automation tool, but not at the enterprise scale. Agentless CA Continuous Delivery Automation tools have a built-in capacity cap because all the task processing occurs on a server. Agent-based CA Continuous Delivery Automation tools distribute the workload, even geographically, and are better able to respond to large workloads. Your enterprise-capable CA Continuous Delivery Automation tool needs to be able to handle thousands of managed nodes, potentially millions of tasks per day, and execute against an array of modern and legacy node types (mainframe, AIX, etc.).

Payload Delivery

An enterprise CA Continuous Delivery Automation tool should be able to move or stage artifacts geographically for deployments. It needs to be able to monitor file transfers, be network-fault tolerant and provide automatic MD checksums to confirm that all data sent is delivered intact and uncorrupted. Your CA Continuous Delivery Automation tool needs to have intelligence built around file transfers so it can alert you, retry or perform a prescribed action upon failure or success. An CA Continuous Delivery Automation tool that cannot perform managed file transfers (MFTs) is not an enterprise-class solution and is not geographically scalable.

Geographic Queued Deployments

An CA Continuous Delivery Automation tool designed for the enterprise provides hierarchal calendaring and scheduling capabilities that are time-zone aware and support regional holidays and corporate blackout dates. For example, you may want to roll out deployments at two a.m. local time. The minute the clock strikes two in any time zone, automated execution of deployments ought to occur without human intervention. You should be able to set global or enterprise-wide blackout dates where no deployments occur. Regional holidays, for example, need to overlay approved deployment schedules. One of the needs in managing releases of large enterprises is to protect you from yourself.

Geographic Data Access

One of the things we consistently hear from large enterprises is the desire to incorporate the comprehensive set of data that accumulates in an CA Continuous Delivery Automation tool into their tailored dashboard, reporting and alerting frameworks. CA Continuous Delivery Automation tools providing data exporting or Web service access to this data may not meet this demand because release management and deployment automation data are hierarchal by nature. Giving a company access to only the raw data places a burden upon the business to transform the data itself.

To efficiently meet the data access needs of the enterprise, an CA Continuous Delivery Automation tool must provide and maintain a flattened reporting database. Enterprises should be able to define the data points of interest and have the CA Continuous Delivery Automation tool populate a table with the requested data. This table should be either regularly appended or refreshed, depending on whether the data represents historical or summary information. A reporting database will allow for easy integration into established corporate dashboards and reporting and alerting mechanisms.

SECTION 4

Load Scalability

Load scalability is what most people think of when the subject of scalability is brought up. An CA Continuous Delivery Automation tool needs to comfortably support expansion and contraction of its resource pool to accommodate heavier and lighter loads. In discussing an earlier dimension, we mentioned the potential for an CA Continuous Delivery Automation tool to perform millions of tasks per day. At an enterprise scale, this can quickly become a reality as you onboard new applications, enable self-help provisioning and deployment services and expand your original use case to include interdependent systems and automate more sophisticated scenarios.

An CA Continuous Delivery Automation solution must support high availability (active/active or active/passive) configurations and be able to handle deployment targets that can come in and out of existence with on-demand environments that are provisioned and de-provisioned as needed. An CA Continuous Delivery Automation solution that supports the highest endpoint-to-server ratio ought to rank at the top of these criteria, because the process of supporting a distribution of servers and their managed nodes can become unruly.

As stated before, agent-based CA Continuous Delivery Automation tools are better at handling high-volume loads because of task execution distribution. That said, make sure the agents of the CA Continuous Delivery Automation tool support job/task recovery and killing of unresponsive commands, and are fault-tolerant in communicating with the server.

SECTION 5

Generational Scalability

Generational scalability, as defined by Wikipedia, has more to do with hardware than software—the idea being that you can upgrade a desktop PC, for example, a year or two later with more RAM or a next-gen graphics card, larger/faster hard drive, etc. From a software perspective, this can be a consideration for what it costs in terms of time and money to regularly upgrade your CA Continuous Delivery Automation tool. Over time the underlying database will have grown, as will the number of managed endpoints or nodes. Migration plans are typically used to upgrade enterprise software, but CA Continuous Delivery Automation vendors, boasting of their continuous delivery prowess, are churning out new versions of their product every one to three months. You can only benefit from frequent product updates if you can migrate your installation to the latest version of the tool in relatively short order.

The CA Continuous Delivery Automation tool you select must be able to provide zero-downtime upgrades of itself. It is ironic that most CA Continuous Delivery Automation tools do not support their own automated deployment. An CA Continuous Delivery Automation tool that cannot auto-update itself is like an accountant with an unbalanced personal budget or a lifeguard who cannot swim. If CA Continuous Delivery Automation vendors are going to practice continuous delivery, then you must demand zero-downtime upgrades of their product. With application architectures changing and the DevOps toolchain expanding, you do not want your CA Continuous Delivery Automation tool to lag behind the technology curve. As your CA Continuous Delivery Automation tool expands in terms of users and applications, the installed components will also expand. This means that upgrading your CA Continuous Delivery Automation tool will be complex and time-consuming, and you will find yourself significantly behind the release cadence of your chosen vendor. CA Continuous Delivery Automation tools must be able to upgrade themselves, including the server(s) and agents. They must consider the deployment of their own product updates as just another automated deployment. At the time of this writing, only one CA Continuous Delivery Automation tool supports zero downtime deployments, which critically enhances its administrative scalability.

SECTION 6

Conclusion

When an CA Continuous Delivery Automation vendor boasts that its tool is scalable, you need to ask what the vendor means by “scalable.” As has been demonstrated, there are five measurable dimensions to scalability, and an CA Continuous Delivery Automation tool that only supports load balancing or high availability is not enough to qualify as an enterprise-class solution. An CA Continuous Delivery Automation solution designed for the enterprise will support all five dimensions. Selecting a one-dimensional or two-dimensional tool will paint your organization into a corner and shortchange the business.

For more information, please visit ca.com/automation

Connect with CA Technologies



CA Technologies (NASDAQ: CA) creates software that fuels transformation for companies and enables them to seize the opportunities of the application economy. Software is at the heart of every business, in every industry. From planning to development to management and security, CA is working with companies worldwide to change the way we live, transact and communicate—across mobile, private and public cloud, distributed and mainframe environments.

Learn more at ca.com.

Copyright © 2018 CA. All rights reserved. All trademarks referenced herein belong to their respective companies. This document does not contain any warranties and is provided for informational purposes only. Any functionality descriptions may be unique to the customers depicted herein and actual product performance may vary.

Some information in this publication is based upon CA's experiences with the referenced software product in a variety of development and customer environments. Past performance of the software product in such development and customer environments is not indicative of the future performance of such software product in identical, similar or different environments. CA does not warrant that the software product will operate as specifically set forth in this publication. CA will support the referenced product only in accordance with (i) the documentation and specifications provided with the referenced product, and (ii) CA's then-current maintenance and support policy for the referenced product.

There are no claims that a CA Technologies product or service has been designed to or may be used by clients/customers to meet regulatory compliance obligations (financial or otherwise).

CA does not provide legal advice. Neither this document nor any CA software product referenced herein shall serve as a substitute for your compliance with any laws (including but not limited to any act, statute, regulation, rule, directive, policy, standard, guideline, measure, requirement, administrative order, executive order, etc. (collectively, "Laws")) referenced in this document. You should consult with competent legal counsel regarding any Laws referenced herein.