RFI Response on Scientific & HPC Software Stewardship by the Department of Energy

Company:
CloudBees, Inc.

Company Address:
3400 W Leigh Street, Richmond, Virginia 23230

Point of Contact:
Hershel Bagley

Federal Register Citation:
86 FFR 60021

From the maker of Jenkins: CloudBees' Software Delivery Automation Platform (SDA) is a Flexible Hybrid-cloud technology that supports software management tools to facilitate deployment and management of services. CloudBees' SDA provides the ability to visualize complex software architectures by standardizing development processes and collecting archifacts that provide those insights and help prevent events like Solarwinds. This ensures compliant standards and best practices are always utilized when building software for the Federal Government. CloudBees' SDA provides measurability, transparency and observability into the development of workflow modeling technologies, mitigating the risk associated with software supply chains for legacy systems and new cloud-based, kubernetes software applications.

Managing interdependent software releases across the Department of Energy's ASCR-funded libraries without an enforceable enterprise standard is complex and often presents certain levels of risk to the process of introducing code into production environments. ASCR-funded software teams, operating in silos, are left to map out and manage their own processes for releasing software, without having the tools or defined process to clearly depict and articulate the requirements in order to integrate with other releases. The result is a lack of visibility, efficiency, cohesion, and standardization in the release orchestration pipelines of ASCR-funded software packages that leads to notable delays and inherited production risks.

The aforementioned delays and inherited production risks can lead to costly errors when software is tested at scale in HPC environments. Allocating time for performance testing on HPC clusters (such as the ones managed by the INCITE program) is expensive and extremely limited. Therefore, it is crucial that development teams have the ability to map out and thoroughly validate the entire software development process for each element of the application prior to undergoing scaled out performance testing. This capability will not only save time and money, but will enable a higher degree of quality assurance for any open-source software release that national labs are providing to the wider commercial industry (CRAY, HPE, Intel, Boeing, Oil & Gas, etc.).

After reviewing the DOE's RFI, we believe that CloudBees' solutions provide capabilities that are directly related to the following areas of the potential scope listed within the Federal Register:

(1) Training: Our industry expertise can provide training on the latest software-development best practices and the use of core software within an enterprise solution.

(2) Infrastructure: CloudBees' SDA platform provides best-in-class, continuously developed infrastructure for software packaging, hosting, testing, and other common capabilities.

(3) Curation: The recently released "Compliance Tool" establishes governance processes and standards to enable resource allocation in the most-effective manner balancing stability with the need to satisfy evolving requirements.

(4) Maintaining situational awareness: The SDA platform gives users and administrators full spectrum visibility of production pipelines--defining, publishing, and communicating understandable information about relevant software and its dependencies across

ASCR-funded teams; collecting information from users and deployment requirements from facilities.

(5) Shared engineering resources: CloudBees can deliver software-engineering resources to assist with maintenance activities of key projects, including the automated triaging of problems from testing and adjusting for new compilers; system software and platform versions; and changing package requirements.

(6) Project support: CloudBees' Technical Success Managers (TSMs) can provide ad hoc support for the continued development of key projects, including enhancing them to function efficiently on new hardware platforms; take advantage of emerging hardware and software technologies; comply with best practices; and otherwise provide high priority features desired by other users.

Through further reading of the DOE RFI, the specific categories that CloudBees can address are as follows:

(1) Software dependencies and requirements for scientific application development and/or research in computer science and applied mathematics relevant to DOE's mission priorities: What software packages and standardized languages or Application Programming Interfaces (APIs) are current or likely future dependencies for your relevant research and development activities? What key capabilities are provided by these software packages? What key capabilities, which are not already present, do you anticipate requiring within the foreseeable future? Over what timeframe can you anticipate these requirements with high confidence? What are the most significant foreseeable risks associated with these dependencies and what are your preferred mitigation strategies? When responding to these questions, please describe the scope of the relevant research and development activities motivating the response.

CloudBees commercial Software Delivery Automation (SDA) provides integrated software solutions that could enhance the Department of Energy's software deployment and integration practices. To keep pace with advanced research capabilities, advanced supercomputing, investment in automated software CI/CD and release solutions are necessary to rapidly translate the Advanced Scientific Computing Research (ASCR) teams' functions and libraries into production-grade components. The resulting software must incorporate the Extreme-Scale Scientific Software Stack's (E4S) robust process of testing and validation, while at the same time be rapidly deployable--which makes a properly adapted and integrated Cloudbees SDA solution essential for DoE software development teams because of its ability to provide governance, compliance, insight and observability into the entire DevSecOps process and the software supply chain.

SDA excels in programs where large complex code footprints are generated across multiple programs and open source communities are leveraged to produce software solutions. CloudBees offers improved processes for automating alternative programming languages (e.g., using Python, R, or Julia); as well as porting to other platforms (e.g., cloud, mobile), automating

user feedback, decreasing complexity of the software delivery lifecycle by streamlining resource and time intensive tasks through automation. Moreover, CloudBees tools ensure the compliance of software builds to make it more robust while adding capabilities to manage deployments in multiple environments with user-support tools and services.

In the case of ASCR, software libraries are heavily utilized for High Performance Computing purposes which are inclusive of both commercial and open source code sets. The validation of open source components and integration with DoE technical solutions is a necessity. The process of building/testing/delivering complex libraries like this can be cumbersome and time consuming. Utilizing SBIR funds to adapt and customize the CloudBees SDA, it would serve as an automation engine to expedite this process by utilizing a CI engine based on Open Source Jenkins. Furthermore SDA includes a release orchestration capability which automates the combination of these components into a release. Specifically SDA collects data and results throughout the build and test processes within a release in order to validate the compliance of an ASCR library before it becomes available in production.

We believe the capability developed by Cloudbees, in collaboration with DoE, will enable the automation of ASCR funded software development and accreditation processes, securely, and at scale. Utilizing CloudBees SDA will bring an effective reduction in the time required to get critical capabilities into the hands of operators, as well as improved speed of the authorization process.

(2) Practices related to the security and integrity of software and data: What strategies and technology do you employ, or intend to employ in the foreseeable future, to ensure the security and integrity of your software and its associated provenance metadata? What capabilities do you provide, or intend to provide in the foreseeable future, to assist users of your software with ensuring scientific reproducibility, recording the provenance of their work products, securing their information, protecting the privacy of others, and maintaining the integrity of their results?

CloudBees is developing a new capability for DevOps pipelines that streamlines governance and security compliance that would benefit the DoE. Some of the features include:

Ensure Full Traceability - *CloudBees ensures that only immutable, approved components, automation and environments are used.*

Integrate Your entire toolchain - *Leverage a vast array of plugins to orchestrate security tools, enabling automated compliance checks and enforcement*

Minimize Human Error - *Automate and manual gates ensure that insecure or non-compliant software does not move downstream.*

Deliver Secure Software Faster - *Use a templated, repeatable process to confidently push new features to production. Integrate approvals, warnings and rejections into the developers' own tools and workflows so they can continuously test code.*

Fix Vulnerabilities Quickly - *Built-in mitigation means any specific feature can be pulled back immediately if crop up-with full traceability of what happened and when.*

Enforce Pipeline Best Practices - *Set fine-grained permissions at the team,user,or file level to easily apply the principle of least privileged. Centrally enforce common security options across multiple controllers with an authentication system, role-based access control, SSO and credential management.*

Transform into Elite-Performing Teams - *Audit and GRC functions become non-events so your teams can focus on innovating and adding value.*

(3) Infrastructure requirements for software development for scientific and high-performance computing: What infrastructure requirements do you have in order to productively develop state-of-the-art software for scientific and high-performance computing? These requirements might include access to testbed hardware, testing allocations on larger-scale resources, hosting for source-code repositories, documentation, and other collaboration tools. What are the key capabilities provided by this infrastructure that enables it to meet your needs? What key capabilities, which are not already present, do you anticipate requiring within the foreseeable future? Over what timeframe can you anticipate these requirements with high confidence? What are the most-significant foreseeable risks associated with this infrastructure and what are your preferred mitigation strategies? When responding to these questions, please describe the scope of the relevant research and development activities motivating the response.

CloudBees Software Delivery Automation (SDA) platform is a dual-use technology for commercial and government that offers the world's first connected, automated, end-to-end software delivery platform that is used today by highly regulated industries like Government, Finance, Insurance and High Tech to deliver world class software securely with the appropriate level of governance and compliance in place through our software. CloudBees enterprise solutions allow organizations connect, automate, orchestrate, and secure DevOps pipelines across development, security, operations, and shared service teams to ensure they are delivering world class software securly. Federal agencies such as the DoD and the IRS are already taking advantage of CloudBees' technology, and with further investigation of DOE needs, we expect to understand ASCR-funded software packages so that we can make significant contributions to the HPC enterprise.

(4) Developing and maintaining community software: How much additional effort is needed to develop and maintain software packages for use by the wider community above the effort needed to develop and maintain software packages solely for use in specific research projects or for internal use? What tasks are the largest contributors to that additional effort? What are the largest non-monetary impediments to performing this additional work? How is any such additional effort currently funded? How does that funding compare to a level of funding needed to maximize impact?

The in-depth experience of CloudBees' of stewarding and commercializing the highly adopted, open-source Jenkins platform would be extremely valuable to DOE and our team would be more than happy to discuss further with ASCR-funded teams once an NDA is in place.

(5) Challenges in building a diverse workforce and maintaining an inclusive professional environment: What challenges do you face in recruiting and retaining talented professionals to develop software for scientific and high-performance computing? What additional challenges exist in recruiting and retaining talented professionals from groups historically underrepresented in STEM and/or individuals from underserved communities? What challenges exist in maintaining inclusivity and equity in the development community for scientific and high-performance computing software? What successful strategies have you employed to help overcome these challenges? What opportunities for professional recognition and career advancement exist for those engaged in developing scientific and high-performance computing software?

Not applicable to CloudBees.

(6) Requirements, barriers, and challenges to technology transfer, and building communities around software projects, including forming consortia and other non-profit organizations: ASCR recognizes that successful software for scientific and high performance computing often has many stakeholders, including academic research activities, research laboratories, and industry. Moreover, while DOE has provided funding for the development of a significant number of foundational software packages within the modern software ecosystem for scientific and high-performance computing, as the complexity of the software ecosystem continues to increase, and number of stakeholders has grown, ASCR seeks to understand how it might encourage sustainable, resilient, and diversified funding and development models for the already-successful software within the ecosystem. Such models include, depending on circumstances that ASCR seeks to better understand, both the private sector and non-profit organizations. Non-profit organizations include both charitable organizations (e.g., those with 501(c)(3) status) and R&D consortia (e.g., those with 501(c)(6) status). What are the important characteristics and components of sustainable models for software for scientific and high-performance computing? What are key obstacles, impediments, or bottlenecks to the establishment and success of these models? What development practices and other factors tend to facilitate successful establishment of these models?

Not applicable to CloudBees.

(7) Overall scope of the stewardship effort: The section labeled Potential Scope, mentioned earlier in the RFI, outlines activities that ASCR currently anticipates potentially including in future programs stewarding the software ecosystem for scientific and high performance computing. Are there activities that should be added to, or removed from, this list? Are there specific requirements that should be associated with any of these activities to ensure their success and maximize their impact?

Any recommendations should be placed here for additional scope that DoE should consider.

(8) Management and oversight structure of the stewardship effort: What do you anticipate will be effective models for management and oversight of the scientific and HPC software ecosystem, and how would that management structure most-effectively interact with DOE and other stakeholders? In addition to DOE, who are the key stakeholders? How can the management structure coordinate with DOE user facilities and others to provide access to relevant testbed systems and other necessary infrastructure?

Defining a standard release for DoE libraries that is powered by the rigor and process defined CloudBees platform mitigates the risk involved with releases of DoE libraries. Specifically call out how these libraries are extremely valuable components of downstream applications and other libraries. A bad release can "poison the well" affecting all dependent software components.

CloudBees could conduct a pilot program on a single software package like MPICH, and then continue to build out an enterprise release orchestration platform that is embedded with automation and integration for DoE DevOps/DevSecOps will unchain ASCR-funded libraries from the manual and error prone processes of application releases and instead allow them to focus on solving science and engineering problems that will benefit our citizens and bolster national security.

(9) Assessment and criteria for success for the stewardship effort: What kinds of metrics or criteria would be useful in measuring the success of software stewardship efforts in scientific and high-performance computing and its impact on your scientific fields or industries?

Not applicable to CloudBees.

(10) Other: What are key obstacles, impediments, or bottlenecks to progress by, and success of, future development of software for scientific and high performance computing? Are there other factors, issues, or opportunities, not addressed by the questions above, which should be considered in the context of stewardship of the ecosystem of software for scientific and high performance computing?

Based on CloudBees' expertise in the space, further considerations that should be addressed include, but are not limited to:

DOE should work to achieve continuous authorization/ATO

CloudBees allows government software teams to deliver trustworthy software delivery pipelines—and for their security teams to more easily grant ATO because they comply with organizational policy.

DOE should deliver secure software faster for beneficiaries of HPC technology

Accelerate software delivery while staying secure at every stage. CloudBees enables automatic security checks and instant mitigation of flawed code—no need to redeploy.

ASCR-funded software packages should pass audits with flying colors

Automate audit reporting and real-time tracking, tracing and approval of software from code through production. Fully automated and secure pipelines give you reports of trustable data with a single click.

DOE can eliminate back-and-forth between teams

Security teams gain transparency and useful insights at every step of the software development life cycle—without waiting for developers to generate and share reports post-development.

Developers can experiment freely—with guard rails

Sleep easy, knowing governance, regulatory and security standards are being met—thanks to a self-service catalog of secure, compliant components, entrance and exit gates and more.

DOE should transform ASCR-funded developers into elite-performing teams

Audit and Governance, Risk, & Compliance functions become non-events so your teams can focus on innovating and adding value.

Hershel Bagley, Senior
CloudBees Account Executive
(703) 624-1030 (c)
hbagley@cloudbees.com