



by, Curating the Production-Grade Cloud-Native
Computing Stack

James Kobielus

October 22nd, 2018



This is a Wikibon Premium Research Note, sponsored by Oracle.

Cloud-Native Opportunities and Challenges

Cloud-native computing presents big opportunities for enterprise application integration. When it's built on open-source platforms, cloud-native computing enables development, deployment, orchestration and management of microservices that can be deployed flexibly and securely across multicloud ecosystems.

Kubernetes is the key to cloud-native computing, but it's far from the only open-source distribution needed for robust deployment and management of containerized microservices. For enterprise IT professionals sifting through the various open-source projects, curating their own cloud-native computing environment can be a daunting challenge. The factors that deepen this challenge include:

- the combinatorial explosion of open-source projects that one would need to test and evaluate in order to build out feature-complete stacks;
- the immaturity of many open-source projects;
- the lack of robust, field-proven deployments of many projects;
- the tricky prospect of integrating these with existing investments in operating systems, databases, and other enterprise IT tools, platforms, and applications; and
- the need to ensure a single point of responsibility for overseeing and managing a computing environment that includes such diverse components as well as legacy IT investments.

The Central Role of CNCF in Cloud-Native Computing

The Cloud Native Computing Foundation (CNCf) is the principal community curating the open-source stack for containerized computing environments. Kubernetes has achieved widespread adoption and rapid maturation through active engagement in CNCf by vendors and users. However, many other component projects under CNCf's umbrella need to come to fruition before the stack can be considered feature-complete, production-ready, or fully mature.

Founded in mid-2015, CNCf is a non-profit [Linux Foundation collaborative project](#) to advance the state-of-the-art for building cloud-native applications and services on open-source software. The organization's [membership](#) includes a wide range of solution providers and enterprise users, and its [governing board](#) includes representatives from Alibaba Cloud, Amazon Web Services, Cisco, Google, IBM, Microsoft, Oracle, Red Hat, and other cloud-native computing solution vendors.

From the start, CNCf—which has strong, broad-based industry participation—has placed a key focus on the Kubernetes project as enabling “open source at the orchestration level, followed by the integration of hosts and services by defining API's and standards through a code first approach to advance the state-of-art of

Curating the Production-Grade Cloud-Native Computing Stack

container-packaged application infrastructure.” CNCF describes “cloud-native” as referring to capabilities that are “container-packaged, dynamically scheduled and microservices-oriented.”

Building on Kubernetes and the Linux Foundation’s separate [Open Container Initiative](#), CNCF’s stated priority is to provide project stewardship that will help the cloud-native computing industry “assemble components to address a comprehensive set of container application infrastructure needs.”

The CNCF stack is a work in progress. In their efforts to build robust solutions on this stack, vendors will have to continually curate through these to determine what’s solid enough to incorporate into production-grade software or cloud services, and very likely may need to complement and extend it all with various blends of open-source and proprietary code from outside CNCF’s purview. Likewise, IT professionals will have to curate through this same stack—and through myriad commercial alternatives that have implemented various combinations of it all—to stand up production-ready cloud-native microservices.

With respect to Kubernetes or any other project in the CNCF stack, even when it has attained “graduated” status, the current version of that project may not support the degree of interoperability, reliability, security, performance, scalability, and other attributes required in every cloud-native computing scenario. Under such circumstances, it may be necessary for vendors and/or users extend, supplement, or otherwise modify the open-source code in order to stand up production-grade cloud-native computing deployments.

How Oracle is Curating the CNCF Stack for Its Customers

Curation works best when vendors play active roles in the CNCF community. Fortunately, CNCF has exceptionally broad participation from solution providers from throughout the cloud-native ecosystem. More than that, many vendors active in developing, testing, and iterating individual projects at various stages of maturity, which gives them in-depth expertise that they bring into their own commercial offerings that incorporate those projects.

Kubernetes has achieved near-ubiquity in the cloud computing arena. As detailed [here](#), Kubernetes has been implemented by vendors of all the leading infrastructure-as-a-service, container, and Linux platforms. Among managed cloud solution providers, [Alibaba Cloud](#), [AWS](#), [Cisco](#), [Microsoft](#), [Google](#), [IBM](#), [Oracle](#), [Red Hat](#), and [VMware](#) all have their respective Kubernetes engines.

Enterprise IT professionals can also rest assured that most cloud-native solution providers deliver CNCF-compliant kernels of Kubernetes and other distributions as those projects progress through “sandbox” and “incubating” to “graduated” status. Indeed, one can measure vendor commitment to any particular CNCF project by the extent to which they:

- Designate representatives to work on those project subgroups as committers, maintainers, approvers, and/or reviewers;
- Play an active role in defining, evolving, enforcing, and defending the vision, values, mission, scope, charter, and code of conduct of the project;
- Provide technical resources—such as artifact repositories, build and test infrastructure, and collaboration tools—to help those groups develop, test, and certify their distributions
- Submit under open-source license code that those groups may incorporate into their artifacts; and
- Incorporate the artifacts from those groups promptly into their own distributions for intensive testing and evaluation.

Among vendors who belong to CNCF, Oracle has been playing a very active role in advancing the cloud-native open-source stack. It recognizes that its engagement with CNCF needs to be top-down, multi-layered, and ongoing. For Oracle, curation of CNCF’s projects into its solution portfolio involves a long-term commitment to integrating them thoroughly into its existing Linux distribution, public cloud services, and enterprise IT solution portfolios.

Curating the Production-Grade Cloud-Native Computing Stack

Though it wasn't one of the original members of CNCF at its founding in 2015, Oracle had already demonstrated support for containers, Kubernetes, and cloud-native computing. Specifically, it had already acquired StackEngine, a container technology company, launched [Oracle Container Cloud Service](#), put its flagship databases and developer tools in the [Docker Store marketplace](#). It had [open-sourced](#) several container utilities, including [Smith](#), a tool for building [microcontainers](#) and for transforming existing Docker containers in to microcontainers; [CrashCart](#), a tool for debugging microcontainers; and [Railcar](#), an Alternative Container Runtime for Rust-based microservices development. And it had [dedicated](#) several engineering teams to Kubernetes development, with a focus on security, networking, and federation.

Key milestones in Oracle's support for CNCF-compliant open-source cloud-native projects include:

- September 2017: Oracle [joined CNCF](#) as a Platinum Member, committing to engaging in the community process that is developing an "open, cloud-neutral, container-native technology stack that avoids lock-in." It designated [Jon Mittelhauser](#), vice president of container native engineering, as Oracle's representative on CNCF's governing board. It also [announced](#) Oracle Container Services for Kubernetes on [Oracle Linux](#), based on Kubernetes version 1.7.4, available for Oracle Linux 7, designed to integrate with the Oracle Container Runtime for Docker. And it open-sourced [Terraform Kubernetes Installer for Oracle Cloud Infrastructure](#), a tool for helping developers easily install, run and manage Kubernetes-based container apps with the extreme performance of bare metal.
- December 2017: Oracle announced that [Oracle Container Services 1.1.8](#) for use with Kubernetes had passed CNCF's new Certified Kubernetes Conformance Program. The release is based on Kubernetes version 1.8.4, is available for Oracle Linux 7, and is designed to integrate with the Oracle Container Runtime for Docker. Later, Oracle Container Services [1.1.9](#) and [1.1.10](#) for use with Kubernetes were released and passed CNCF's Certified Kubernetes Conformance Program.
- In 2018 so far: Oracle has added [support for CNCF's open serverless standards](#)—[Cloud Events](#), [Serverless Framework](#), and [OpenCensus](#)—to its own [Fn](#) open-source serverless project. Of more immediate interest for customers interested in running Kubernetes in Oracle Cloud, the vendor [has added](#) several new production-grade Kubernetes Features to [Oracle Container Engine](#) on Oracle Cloud Infrastructure (OCI).

But the vendor's most important announcement this year so far in this context was in October at OpenWorld 2018. The vendor [announced Oracle Linux Cloud Native Environment](#), which is a curated set of open-source, enterprise-grade, CNCF-compliant projects. With Oracle Linux Cloud Native Environment, the vendor selects projects from CNCF and makes them available in preview mode before adding them to the Oracle supported stack.

Oracle Linux Cloud Native Environment includes all the components discussed in Table 2.

Curating the Production-Grade Cloud-Native Computing Stack

Container infrastructure	<p>The stack includes tools for creating and provisioning Open Container Initiative (OCI)-compliant containers with the Oracle Container Runtime for Docker package available for Oracle Linux 7 on both the x86_64 and Arm architectures. To provide additional security and isolation of workloads, Oracle has adopted Kata Containers, an OpenStack Foundation project. Oracle is using Kata Container software to deliver the framework for creating lightweight virtual machines that can easily plug into a container ecosystem. A combination of Intel's Clear Container initiative and the Hyper runV project, Kata Containers offer additional levels of security while maintaining the development and deployment speed of traditional containers. Kata Containers are available as a developer preview with Oracle Linux.</p>
Orchestration and management	<p>The stack includes Oracle Container Services for use with Kubernetes as an extension to Oracle Linux, based on the upstream Kubernetes project and released under the CNCF Kubernetes Certified Conformance program. Oracle Container Services for use with Kubernetes simplifies the configuration and setup of Kubernetes with support for backup and recovery. It was developed for Oracle Linux and integrates with Oracle Container Runtime for Docker to provide a comprehensive container and orchestration environment for the delivery of microservices and next-generation application development. CRI-O, an implementation of the Kubernetes CRI (Container Runtime Interface) to enable using Open Container Initiative compatible runtimes, is available in preview. CRI-O allows you to run containers directly from Kubernetes without any unnecessary code or tooling. As long as the container is OCI-compliant, CRI-O can run it, cutting out extraneous tooling and allowing containers to do what they do best: fuel your next-generation cloud native applications.</p>
Networking	<p>The stack includes Flannel, an open-source tool that provides the overlay network used in Oracle Container Services for use with Kubernetes today and simplifies container-to-container networking. The Container Network Interface (CNI) project currently incubating under CNCF seeks to simplify networking for container workloads by defining a common network interface for containers. The CNI plugin is available as a developer preview. Coming soon additional features like Calico will enable customers to define fine-grained connection policies to further improve container and virtual machine network security.</p>
Storage	<p>The stack includes several storage projects that are associated with the CNCF foundation for use with Kubernetes, including a plugin for Gluster Storage for Oracle Linux Release 3.12. The future of storage integration will be provided through the use of a new plugin referred to as the Container Storage Interface (CSI) which was released in alpha beginning with Kubernetes 1.9. This new plugin will adhere to a standard specification and allow storage vendors to manage their plugins against their own timelines versus alignment with upstream Kubernetes releases. The alpha CSI plugin is available as a developer preview.</p>
Observability and diagnostics	<p>The stack includes CNCF graduated project Prometheus for monitoring container environments. It provides time-series dimensional data, powerful query tools and alerting features to improve visibility across the environment. In addition, integration with 3rd party "exporters"—such as Fluentd, which, like Prometheus, is available in preview—which allows users to collect additional data and turn it into a metric in Prometheus.</p>

Continuous Integration / Continuous Delivery:	The stack includes Jenkins X, which is available in preview, and is a CNCF project that rethinks how developers should interact with CI/CD in the cloud with a focus on making development teams more productive through automation, tooling and DevOps best practices.
---	---

Table 2: Oracle Linux Cloud Native Environment

Oracle is positioning Oracle Linux Cloud Native Environment as having the differentiated value propositions discussed in Table 3.

Openness	Oracle Linux Cloud Native Environment is a CNCF-compliant open source stack offers freedom from lock-in and allows customers to stay up to date with the latest technologies.
Choice	Oracle Linux Cloud Native Environment enables customers to deploy cloud-native apps where they want: within on-premises private clouds, in popular public clouds, including Oracle Cloud. Customers can use Oracle VM VirtualBox to develop cloud-native apps on the desktop of their choice and then deploy them to the cloud.

Trust	Oracle Linux Cloud Native Environment is tested by Oracle in its public cloud first to make sure everything is working well. The vendor selects projects from CNCF and makes them available in preview mode before adding them to the Oracle supported stack. Oracle then shares useful templates and workflows that it creates in this testing process with its customers. The stack's components can be easily deployed and have been tested by Oracle for interoperability and production readiness. The stack comes with Oracle worldwide enterprise-grade support, giving customers the confidence to run the business on an open-source environment at scale. Oracle provides collaborative support tickets across the stack, encompassing Oracle Linux Cloud Native Environment, Oracle Linux, Oracle Cloud or Oracle infrastructure (if customers choose to deploy there) as well as encompassing other elements of the Oracle software stack including the Oracle Database and Oracle Applications. Oracle is offering up to 3,500 free hours on Oracle Cloud to developers that would like to use its cloud for customers to run the curated CNCF-compliant stack in their development environment.
Simplicity	The Oracle Linux yum server provides simple access to Oracle Linux Cloud Native Environment software and other Linux developer preview software. Oracle VM VirtualBox helps customers get started with Oracle Linux Cloud Native Environment quickly and deploy it on their desktop of choice or deploy it to the cloud. Support for the Oracle Linux Cloud Native Environment is included with an Oracle Linux Premier support subscription at no additional cost. Components available in preview are made available via Oracle Linux yum server or Oracle Container Registry. The stack includes Oracle as a single source of support, as well as pretested templates and workflows simplify configuration and deployment.

Table 3: Key Differentiated Value Propositions of Oracle Linux Cloud Native Environment

Taking the Next Step with Curated Cloud-Native Stacks

To accelerate their cloud-native journeys, enterprise customers should explore curated cloud-native stacks such as Oracle Linux Cloud Native Environment. CNCF-compliant cloud-native stacks with enterprise-grade support offer fast, simple approaches for deploying production-grade, feature-complete cloud-native computing environments to support continuous integration and deployment of containerized and orchestrated microservices.

Wikibon supports Oracle's focus on CNCF projects that meet the quality and availability needs of enterprises deploying mission critical systems. Wikibon also applauds Oracle's decision to work with the community and to deliver only CNCF-compliant solutions. And we are impressed with the vendor's differentiated value propositions for Oracle Linux Cloud Native Environment: openness, choice, trust, and simplicity.

With its community engagement and curated software stack, Oracle has raised the high bar for providers of enterprise-grade cloud-native software solutions.

Former Wikibon analyst



James Kobielus

@jameskobielus

james.kobielus@wikibon.org