## ***Orchestration***

* **Overview**

Orchestration is the automated configuration, management, and coordination of computer systems, applications, and services. Orchestration helps IT to more easily manage complex tasks and workflows.

IT teams must manage many servers and applications, but doing so manually isn’t a scalable strategy. The more complex an IT system, the more complex managing all the moving parts can become. The need to combine multiple automated tasks and their configurations across groups of systems or machines increases. That’s where orchestration can help.

[Automation](https://www.redhat.com/en/topics/automation) and orchestration are different, but related concepts. Automation helps make your business more efficient by reducing or replacing human interaction with IT systems and instead using software to perform tasks in order to reduce cost, complexity, and errors.

In general, automation refers to automating a single task. This is different from orchestration, which is how you can automate a process or workflow that involves many steps across multiple disparate systems. When you start by building automation into your processes, you can then orchestrate them to run automatically.

IT orchestration also helps you to streamline and optimize frequently occurring processes and workflows, which can support a [DevOps approach](https://www.redhat.com/en/topics/devops) and help your team [deploy applications more quickly](https://www.redhat.com/en/topics/automation/what-is-deployment-automation).

* Best Container Orchestration Tools and services

**1. Kubernetes**

**2. Openshift**

**3. Hasicorp Nomad**

**4. Docker Swarm**

**5. Rancher**

**6. Mesos**

**7. Google Cloud Run**

**8. AWS Elastic Kubernetes Service (EKS)**

**9. Amazon EC2 Container Service (ECS)**

**10. AWS Fargate**

**11. Azure AKS Service**

**12. Azure Managed Openshift Service**

**13. Azure Container Instances**

**14. Digital Ocean Kubernetes Service**

**15. Red Hat OpenShift Online**

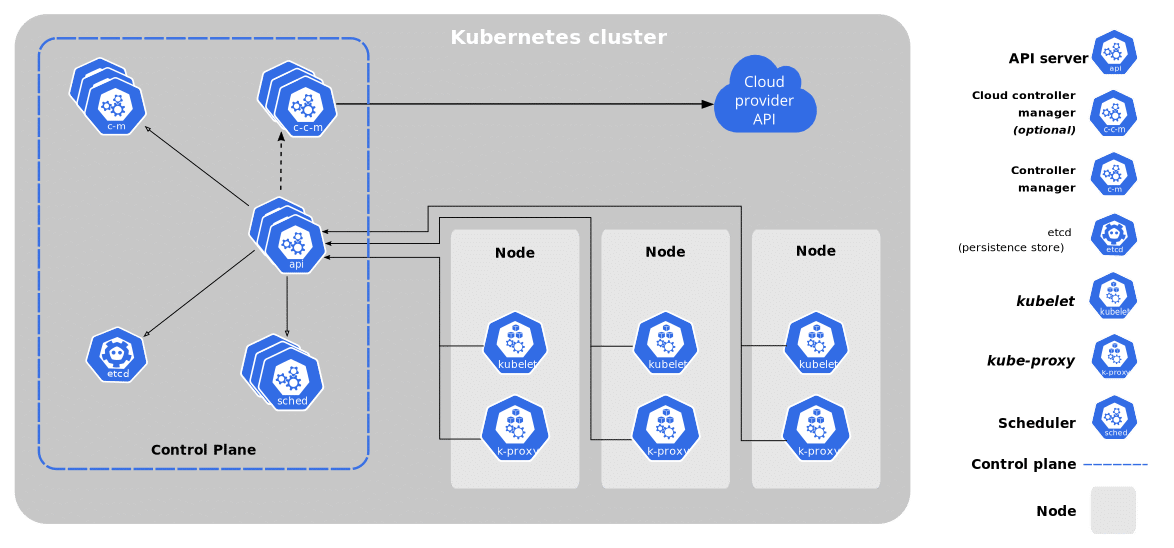
**16. Linode Kubernetes Engine**

**1.Kubernetes**

[Kubernetes](https://devopscube.com/kubernetes-deployment-tutorial/) is an open-source, out-of-the-box container orchestration tool. It comes with an excellent scheduler and resource manager for deploying highly available containers more efficiently.

Kubernetes has become the de facto container orchestration tool for many organizations.

**Survey Report:**According to CNCF’s [Cloud Native Landscape](https://landscape.cncf.io/), there are more than 109 tools to manage containers, but 89% are using different forms of Kubernetes.

The kubernetes project is maintained by the cloud-native foundation with contributors worldwide. Contributors include big organizations to individual open source developers. Here is the high-level architecture of Kubernetes. 

**2. Openshift**

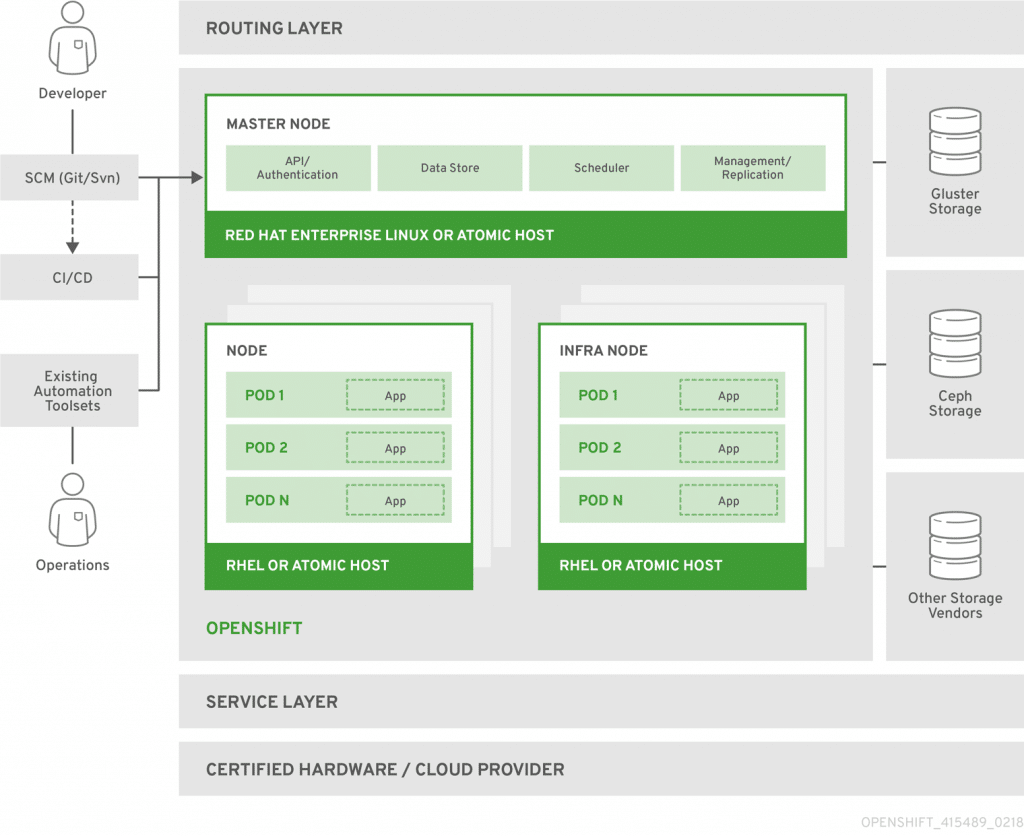
Openshift is built on top of kubernetes.

There is a community as well as an enterprise version of Openshift.

Redhat maintains the Openshift project. Both open-source (openshift origin) and enterprise version ([openshift container platform](https://www.openshift.com/products/container-platform/" \t "_blank)).

Along with core Kubernetes features, it offers container management and orchestration out-of-the-box features.

Here is the high-level overview of Openshift.



**3. Hasicorp Nomad**

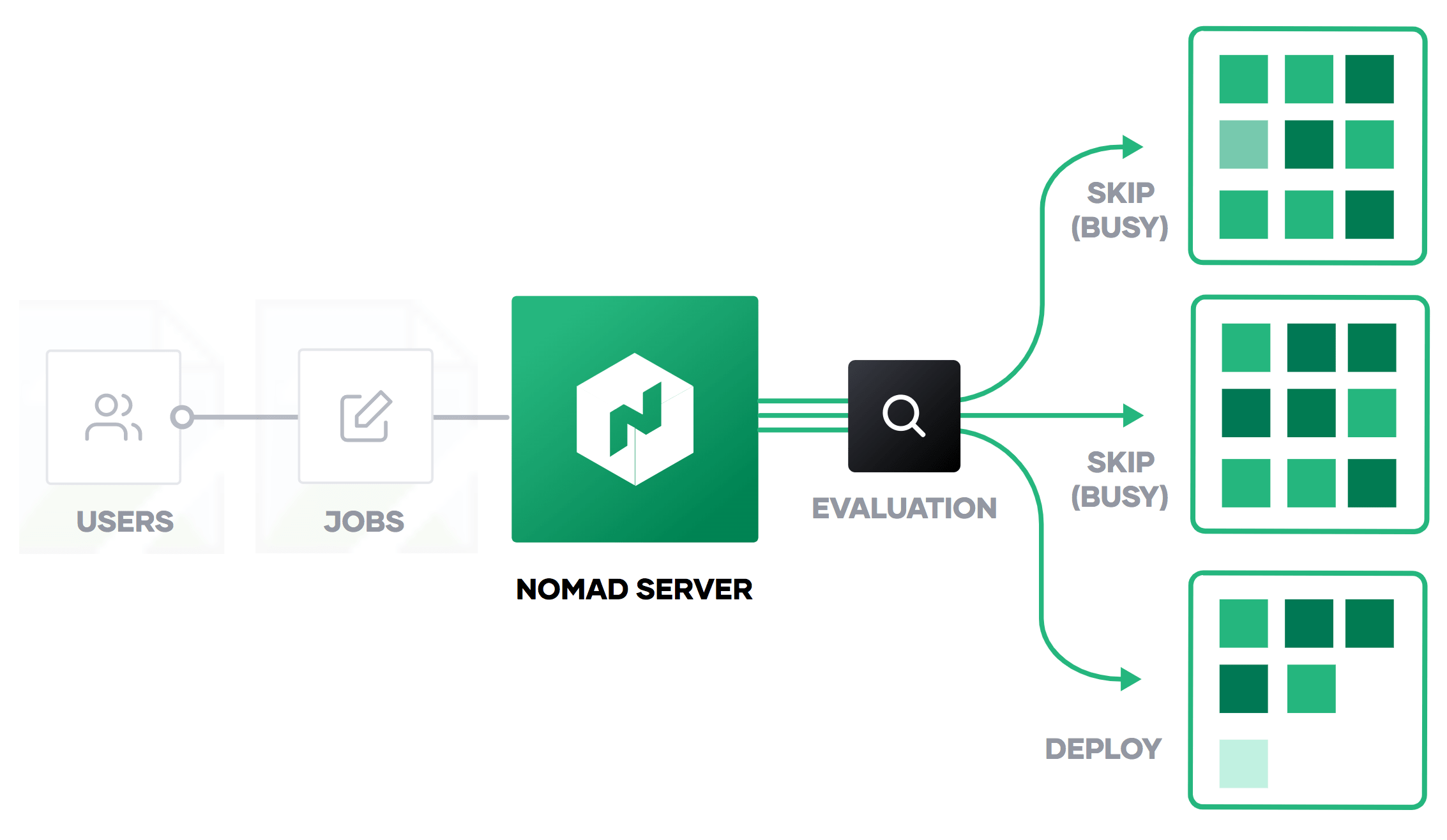
Nomad is an orchestration platform from Hashicorp that supports containers. It shares a similar philosophy of kubernetes in managing applications at scale.

However, Nomad supports container and non-container workloads.

Nomad comes with good integration of other Hashicorp tools like [Consul](https://devopscube.com/setup-consul-cluster-guide/), [Vault](https://devopscube.com/setup-hashicorp-vault-beginners-guide/), and [terraform](https://devopscube.com/setup-google-provider-backend-terraform/).

Primary use-cases for Nomad are,

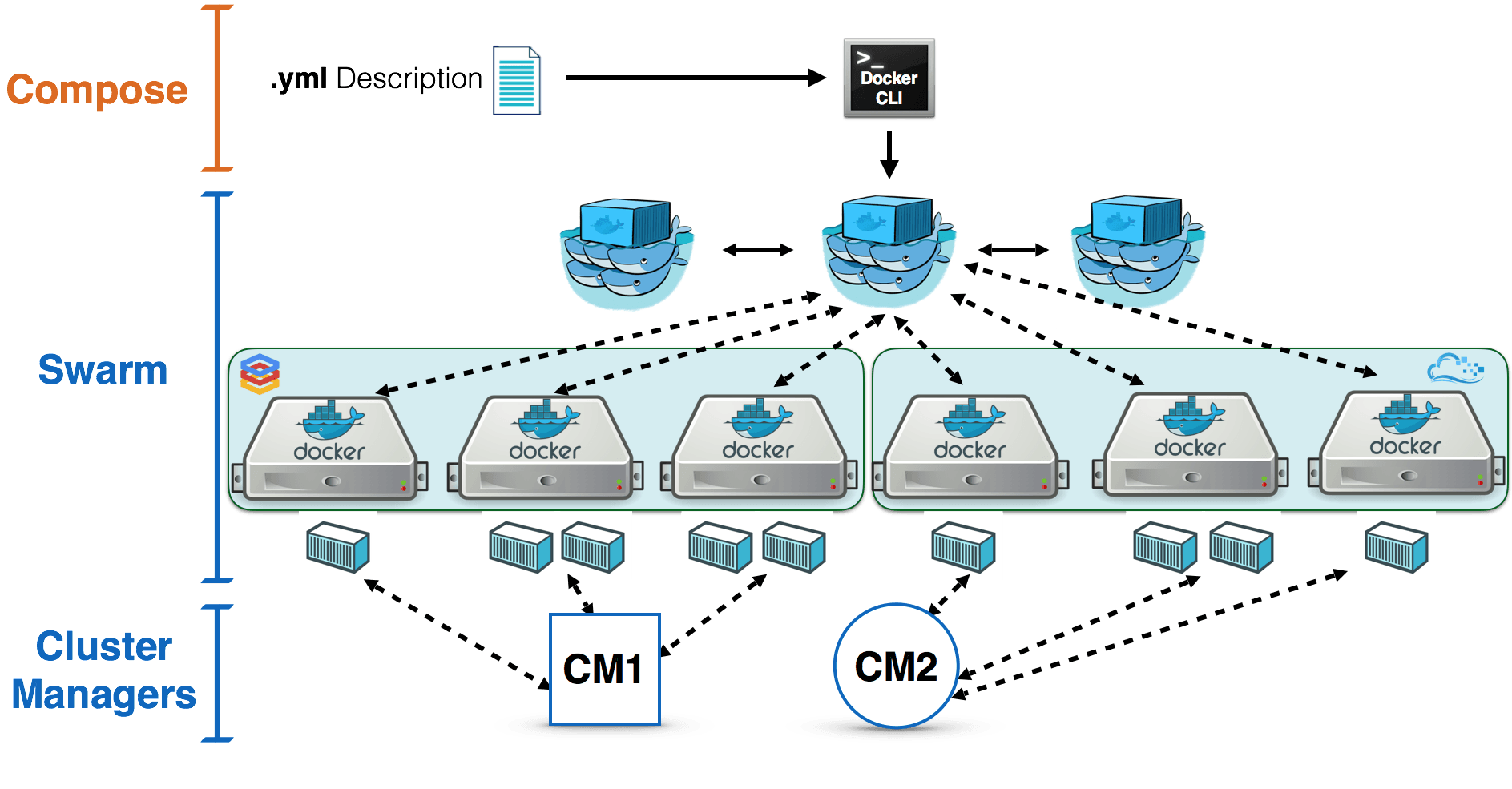
1. Container Orchestration
2. Non-containerized application orchestration.
3. Automated service networking with Consul.

You can read about how Cloudflare uses Nomad for their [dynamic task scheduling system](https://blog.cloudflare.com/how-we-use-hashicorp-nomad/)

1. **Docker Swarm**

The Docker ecosystem consists of tools from development to production deployment frameworks. In that list, [docker swarm](https://devopscube.com/how-to-setup-and-configure-docker-swarm-cluster/) fits into cluster management. A mix of docker-compose, swarm, overlay network, and an excellent service discovery tool such as etcd or consul can be used for managing a cluster of Docker containers.

[Docker swarm](https://devopscube.com/docker-tutorial-getting-started-with-docker-swarm/) is still maturing in terms of functionalities compared to other open-source container cluster management tools. Considering the vast docker contributors, it won’t be long for the docker swarm to have all the best functionalities other tools possess. Docker has documented a good production plan for using the docker swarm in production.

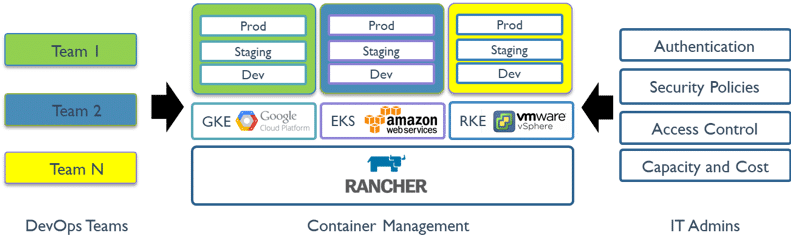


1. **Rancher**

Rancher is an opensource container orchestration tool. At the core, the rancher uses kubernetes as the container orchestrator.

Rancher offers the following functionalities.

1. Centralized Cluster Provisioning that supports on-prem, cloud, and edge.
2. Streamlined Kubernetes Operations by controlling cluster operations from a single console.
3. Centralized Kubernetes Security through centralized user policies.
4. Intuitive Workload Management using native [kubernetes API](https://devopscube.com/kubernetes-api-access-service-account/" \t "_blank) or kubectl utility.
5. Integrated Monitoring and Logging using [Prometheus](https://devopscube.com/setup-prometheus-monitoring-on-kubernetes/), Fluentd, and [Grafana](https://devopscube.com/integrate-visualize-prometheus-grafana/).
6. Supports management for Amazon EKS clusters &[Google Kubernetes Engine (GKE)](https://devopscube.com/setup-kubernetes-cluster-google-cloud/)
7. Global Application Catalog to make application installation and upgrade easier.

[](https://devopscube.com/wp-content/uploads/2021/07/platform.png)

Types of orchestration

1.Classical Orchestra

2.Orchestral Pop

3.Orchestral Hip Hop

4.Electronic Orchestra

5.Orchestral Jazz

**1.Classical Orchestra**

As mentioned above, the most recognised genre of orchestral music is classical orchestra music. Classical music follows traditional forms of making music. Strictly speaking, classical music describes music composed during the Classical period of 1750-1820. The most famous composers of this time were Mozart, Bach and Beethoven.

**2.Orchestral Pop**

Orchestral pop is, quite simply, pop music that’s performed by an orchestra. Personally, the sub-genre reminds us of the most recent James Bond theme tunes – including Adele’s Skyfall, Sam Smith’s Writing on the Wall and Billie Eilish’s No Time to Die; but if you’re not a fan of the spy franchise, you may be more familiar with pop songs such as Clean Bandit’s Rather Be and Ariana Grande’s ghostin.

**3.Orchestral Hip Hop**

When hip hop rhythms are created by an orchestra, the song can be categorised as orchestral hip hop. The sub-genre has been growing ever since the 90s, and has blessed us with anthems such as Coolio and L.V.’s Gangsta's Paradise and Kanye West’s Flashing Lights.

4.**Electronic Orchestra**

The electronic orchestra genre consists of compositions that are either performed by an orchestra and later manipulated electronically, or combine orchestral instruments with electronic instruments. One of the best examples of electronic orchestra music is Daft Punk’s score for Disney’s Tron Legacy that was created with an 85-piece orchestra and electronic instruments

. **5.Orchestral Jazz**

# The genesis of orchestral jazz dates back to New York City’s 1920s jazz scene. This structured style of jazz is created when an orchestra embraces the rhythms of New Orleans jazz whilst sticking to their European practices. This genre is comparable to the sounds of another sub-genre of jazz, big band jazz.

# Workflow engine

# A **workflow engine** is a [software application](https://en.wikipedia.org/wiki/Software_application) that manages [business processes](https://en.wikipedia.org/wiki/Business_process). It is a key component in [workflow technology](https://en.wikipedia.org/wiki/Workflow_technology) and typically makes use of a [database server](https://en.wikipedia.org/wiki/Database_server).

A workflow engine manages and monitors the state of activities in a [workflow](https://en.wikipedia.org/wiki/Workflow), such as the processing and approval of a loan application form, and determines which new activity to transition to according to defined processes (workflows).[[1]](https://en.wikipedia.org/wiki/Workflow_engine#cite_note-1) The actions may be anything from saving an application form in a [document management system](https://en.wikipedia.org/wiki/Document_management_system) to sending a reminder e-mail to users or escalating overdue items to management. A workflow engine facilitates the flow of information, tasks, and events. Workflow engines may also be referred to as Workflow Orchestration Engines.[[2]](https://en.wikipedia.org/wiki/Workflow_engine#cite_note-2)

Workflow engines mainly have three functions:

* Verification of the current process status: Check whether it is valid executing a task, given current status.
* Determine the authority of users: Check if the current user is permitted to execute the task.
* Executing condition script: After passing the previous two steps, the workflow engine executes the task, and if the execution successfully completes, it returns the success, if not, it reports the error to trigger and roll back the change.[[3]](https://en.wikipedia.org/wiki/Workflow_engine#cite_note-3)

A workflow engine is a core technique for task allocation software, such as [business process management](https://en.wikipedia.org/wiki/Business_process_management), in which the workflow engine allocates tasks to different executors while communicating data among participants. A workflow engine can execute any arbitrary sequence of steps, for example, a healthcare data analysis.[[4]](https://en.wikipedia.org/wiki/Workflow_engine#cite_note-hf2010-4)

See also[[edit](https://en.wikipedia.org/w/index.php?title=Workflow_engine&action=edit&section=1)]

* [Business rules engine](https://en.wikipedia.org/wiki/Business_rules_engine)
* [Business rule management system](https://en.wikipedia.org/wiki/Business_rule_management_system)
* [Comparison of BPEL engines](https://en.wikipedia.org/wiki/Comparison_of_BPEL_engines)
* [Inference engine](https://en.wikipedia.org/wiki/Inference_engine)
* [Java Rules Engine API](https://en.wikipedia.org/w/index.php?title=Java_Rules_Engine_API&action=edit&redlink=1)
* [Rete algorithm](https://en.wikipedia.org/wiki/Rete_algorithm)
* [Ripple down rules](https://en.wikipedia.org/wiki/Ripple_down_rules)
* [Semantic reasoner](https://en.wikipedia.org/wiki/Semantic_reasoner)
* [Business Process Execution Language](https://en.wikipedia.org/wiki/BPEL)
* [Production system](https://en.wikipedia.org/wiki/Production_system_(computer_science))
* [Workflow management system](https://en.wikipedia.org/wiki/Workflow_management_system)
* [Joget Workflow](https://en.wikipedia.org/wiki/Joget_Workflow)
* [Conductor (software)](https://en.wikipedia.org/wiki/Conductor_(software))

Characteristics of orchestration

* Support for Several Types of Virtual Machines. A cloud orchestration platform should be able to support multiple types of virtual machines.
* Reusable Templates.
* DevOps Tools.
* Access Control.
* Single Pane View.
* Multi-Cloud Support.
* Extensibility.