



Orchestral.ai

F A Q

# Orchestral.ai

*Conquering Complexity in Enterprise IT*



```
elif operation == "MIRROR Y":
    mirror_mod.use_x = False
    mirror_mod.use_y = True
    mirror_mod.use_z = False
elif operation == "MIRROR Z":
    mirror_mod.use_x = False
    mirror_mod.use_y = False
    mirror_mod.use_z = True

#selection at the end -add back the deselected
mirror_ob.select= 1
modifier_ob.select=10 10 1 0
bpy.context.scene.objects.active = modifier_ob
print("Selected" + str(modifier_ob)) #modifier
#mirror_ob.select = 0
time = bpy.context.selected_objects[0]
time.data.objects[0].name = "selected"

@classmethod
def poll(cls)
    return c
```



## Why Orchestral.ai?

Orchestral's mission is to enable IT infrastructure & operations teams to more effectively manage the complex mission critical processes that their organizations depend upon for day-to-day operations. We accomplish this today with our Composer platform as an enterprise-grade version of the open source StackStorm project with extensions, enhancements, integrations, training and support not available from the open source community.



## What is Composer?

Based on the open source StackStorm project, Composer is a powerful automation and orchestration platform that enables IT teams to automate their operations and reduce the complexity of managing large-scale hybrid cloud infrastructure. Learn more about Composer - [here](#).



## When do I deploy Composer?

Composer can be deployed whenever an IT team is struggling to manage the day-to-day operations of large-scale infrastructure. Whether it be manual tasks and interventions or stove piped workflows that are taking weeks to finish, Composer can help you automate your infrastructure and pave a path for a more efficient and agile business.



## How do I use Composer?

Composer is an if-this-then-that (IFTTT) sensor and rule based workflow engine that hundreds of enterprise infrastructure integrations that enable users to connect a multitude of disparate tools, devices and systems to address a wide range of use cases. Composer provides a graphical design canvas where users can visually construct workflows in a low-code/no-code interface that enables efficient management of complex multi-step processes.



## How do I deploy Composer?

The typical best practice approach is to deploy Composer in a High Availability (HA) manner. This would involve 2 Linux VM's of Ubuntu 18 or Centos 8, clustered together to ease the process of HA deployment. This would be followed by integrating Composer with the existing infrastructure via integration "Packs" and the utilize the workflow orchestrator to begin automating workflows.



## What is event driven automation?

An important differentiating feature of Composer is its ability to deploy sensors for every automation Pack and every corresponding component in the infrastructure. These Sensors in turn trigger (or activate) multiple Actions within the Composer platform. This could be one small Action or hundreds of workflow Actions strung together. Composer also integrates with a wide variety of infrastructure and application end points to trigger actions and workflows



## What is the difference between StackStorm and Composer?

StackStorm provides hundreds of open source packs that form the underlying foundational platform for Orchestral's Composer. Composer provides enterprise-grade efficiency improvements as well as enhancements for scalability, RBAC support and the ability to roll back complex workflows in the event of task failure. This is in addition to several other improvements to the UI/UX, including an innovative Zero-Touch Support (ZTS) Pack. Learn more about the differences between Composer and StackStorm - [here](#).



## What are the most popular Packs used in Composer workflows?

The most popular Composer Packs fall into (5) categories that encompass the data center and business systems automation:

1. ChatOps packs supporting Slack, Teams, and PagerDuty.
2. Infrastructure target systems supporting the cloud orchestration such as AWS, Azure, and GCP. Virtualization orchestration such as VMware, Nutanix, and OpenStack. Server, storage and network automation such as Dell, EMC, and Cisco.
3. Datacenter Security packs supporting Palo Alto, Checkpoint and Fortinet.
4. IT Service Management (ITSM) packs supporting ServiceNow, Elasticsearch, Splunk.
5. SDLC packs supporting CICD, Github and Jenkins.



## Does Orchestral provide professional services & training?

Yes, our Professional Services team provide a number of customer success services including building custom workflows, as well as onboarding and training for Composer as well as basic guidance on working with open source StackStorm. Our goal is to assist enterprise infrastructure teams to efficiently build their own workflow development process as they come up to speed. The learning curve is short due to the intuitive nature of the Composer UI.



## How scalable is Composer?

The Composer architecture is based on microservices. Because of this, the Composer platform is horizontally scalable with additional enhancements to support a number of high-availability deployment scenarios. And, scalability configuration/reconfiguration does not require reinstalling the platform.



## What is the user interface for Composer?

Composer utilizes Apache Web Server with a web user interface, along with Visual Workflow Designer. The Visual Workflow Designer enables users to easily create new workflows and modify existing workflows on a drag and drop canvas to maximize ease of use.



## How does **Orchestral** price the **Composer** solution?

Orchestral.ai has two pricing tiers. The first tier is our Premier offering which includes the Composer Platform and Premier Packs that we provide. The second tier is for Enterprise Level Support for users of the StackStorm open source packs.



## What are the differences between the Standalone & HA version of Composer?

The Standalone version of Composer is a single instance, while the High Availability Deployment Version (HA) has multiple instances to ensure that another (secondary) instance will activate in the event the primary instance fails. This ensures a zero downtime installation.



## How many concurrent workflows can a single instance of Composer support?

That depends on the complexity of the workflow requirements. Composer can support 1000's of simultaneous workflows. Composer's workflows execute very quickly, most often in a matter of seconds. Typically, latency is introduced when various infrastructure elements take time to respond and therefore the workflow must then wait for the infrastructure elements to complete their responses. However, Orchestral has very specific enhancements that address this issue to even further extend Composer's scalability.



## How can I see a Composer demo?

The most effective demos are based on real-world use cases in your environment. There are many use cases we can show such as infrastructure provisioning and deprovisioning, event notification and auto remediation, automated patch management, and dozens more. If you have 10-15 minutes, we can show you a quick, high level demonstration of our capabilities. For a more complex demo around your specific use case(s), please contact our Client Development Team via [info@orchestral.ai](mailto:info@orchestral.ai).

## About Us

Orchestral's mission is to enable IT infrastructure & operations teams to more effectively manage the complex mission critical processes that their organizations depend upon for day-to-day operations. We accomplish this today with our Composer platform as an enterprise-grade version of the open source StackStorm project with extensions, enhancements, integrations, training and support not available from the open source community.



**Orchestral.ai**  
Conquering Complexity in Enterprise IT

Contact us at [info@orchestral.ai](mailto:info@orchestral.ai) to schedule a demo and discover the ways we can help you conquer complexity.

©2023 Orchestral.ai, Inc. All rights reserved. Orchestral.ai and the Orchestral.ai logo are trademarks or registered trademarks of Orchestral.ai, Inc. in the United States and/or other countries. All other names are the property of their respective owners. For additional information on Orchestral.ai Trademarks please see <http://www.orchestral.ai/company/legal/trademarks>. Specifications and product availability are subject to change without notice.