

Extensions Framework & Orchestrate Anything

About Me

- **Harikrishna Patnala**

- PMC and Committer @ Apache CloudStack project
- Born and brought up in CloudStack
- Lead Software Engineer @ ShapeBlue



Recap – CCC 2024



External Compute Deployment in Apache CloudStack

Harikrishna Patnala
Alex Mattioli



November 20 - 22, 2024 | Madrid, Spain

Why Extensions?

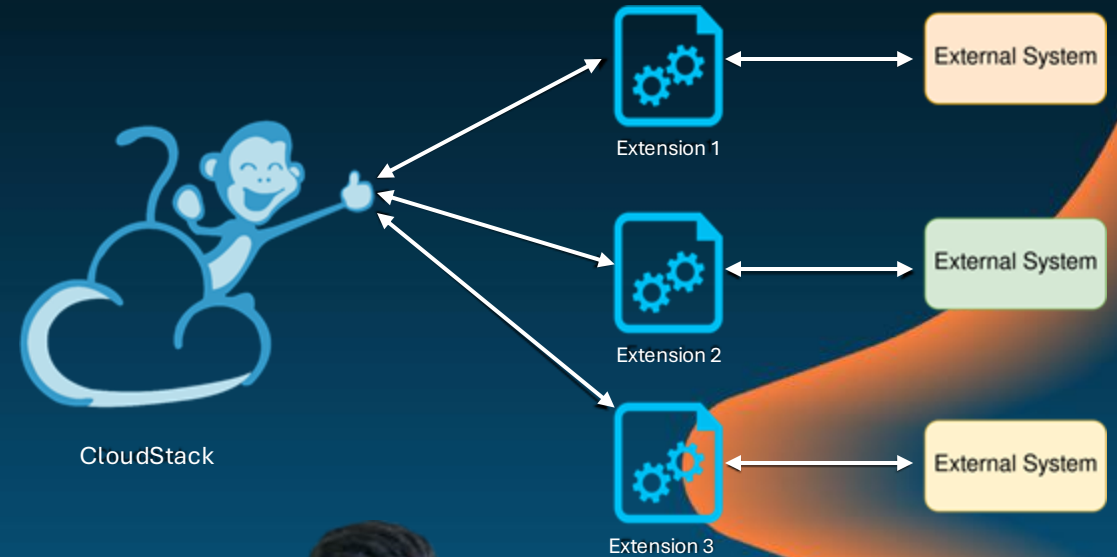
- Plug in External Logic Easily
 - Integrate custom scripts or tools directly into CloudStack workflows
 - Ideal for operators and developers outside the core project
- Decouples external logic from CloudStack core
- Integrations can be done easily

What can be Achieved?

- Can integrate new Instance provisioners or hypervisors
 - Proxmox
 - Hyper-V
 - Firecracker
 - MaaS
 - Baremetal
- Define custom actions like
 - Snapshots
 - Clone operations
 - Backups
- Can integrate new Network Extensions

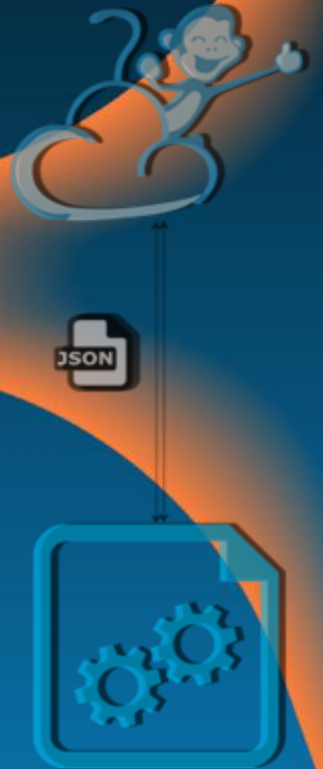
Extensions Framework

- Integrates external systems and workflows
- An executable binary or script in any programming language that acts as a bridge between CloudStack and the external system
- Available from 4.21.0

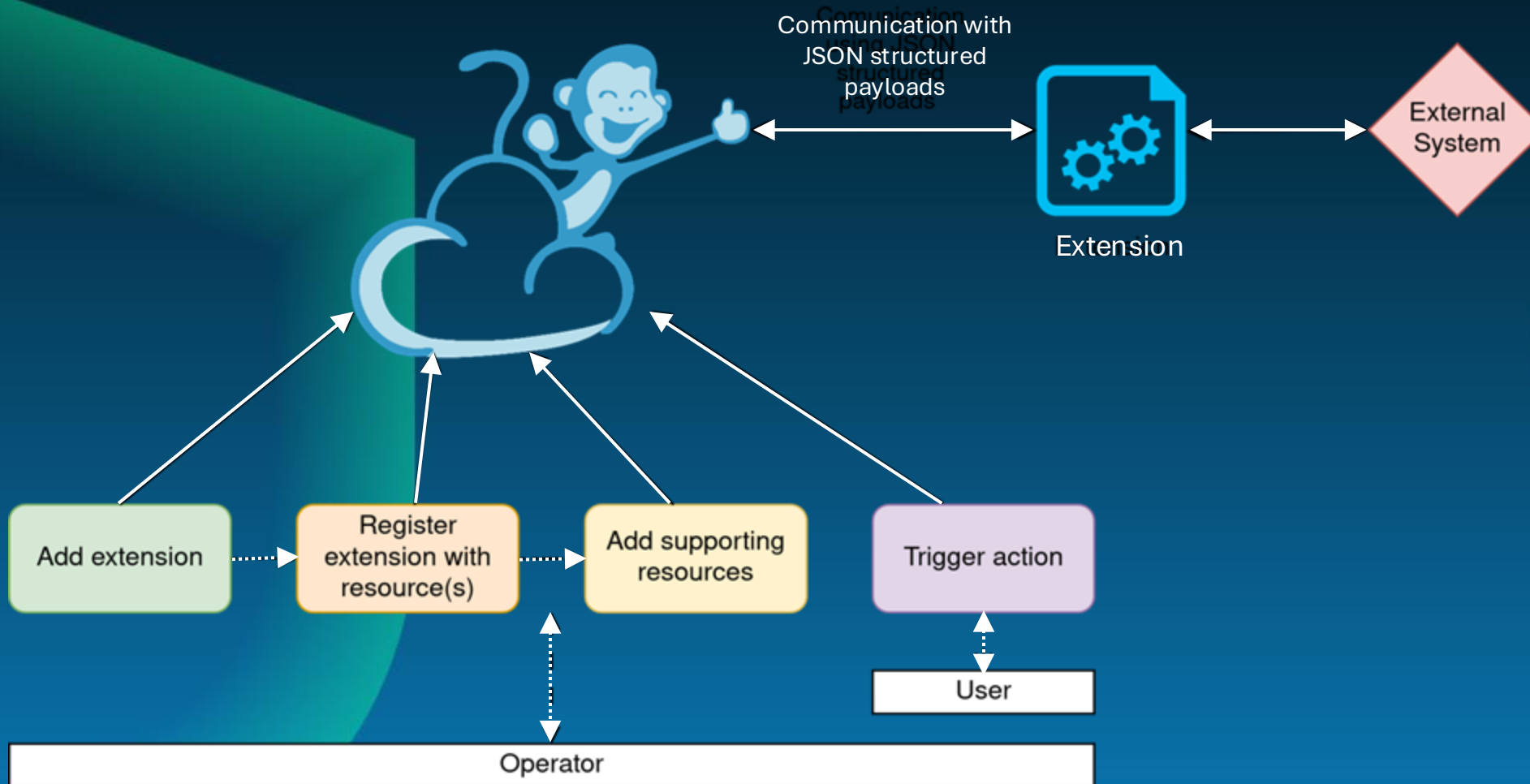


Extensions Framework

- Extensions of different types can be defined.
 - Currently **Orchestrator** type is supported.
- Communication using **JSON** structured payload
- Ability to define **custom actions**
- Extension **binary or script file(s)** will be placed at
`/usr/share/cloudstack-management/extensions/<EXTENSION_NAME>`
- Extension data will be stored at
`/var/lib/cloudstack/management/extensions/<EXTENSION_NAME>`

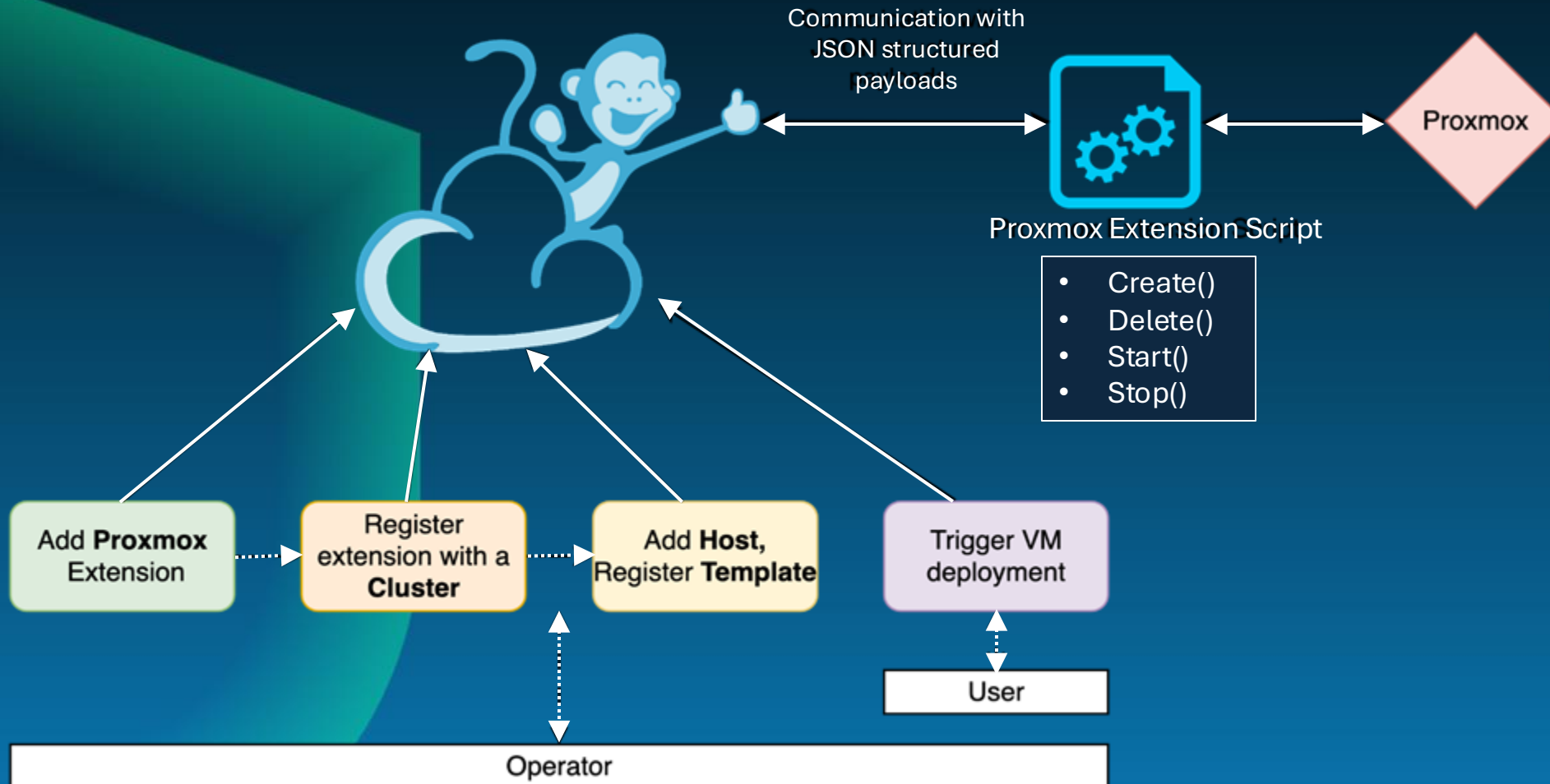


Extension Framework - Workflow



Extension Framework

Workflow Example



Custom Actions

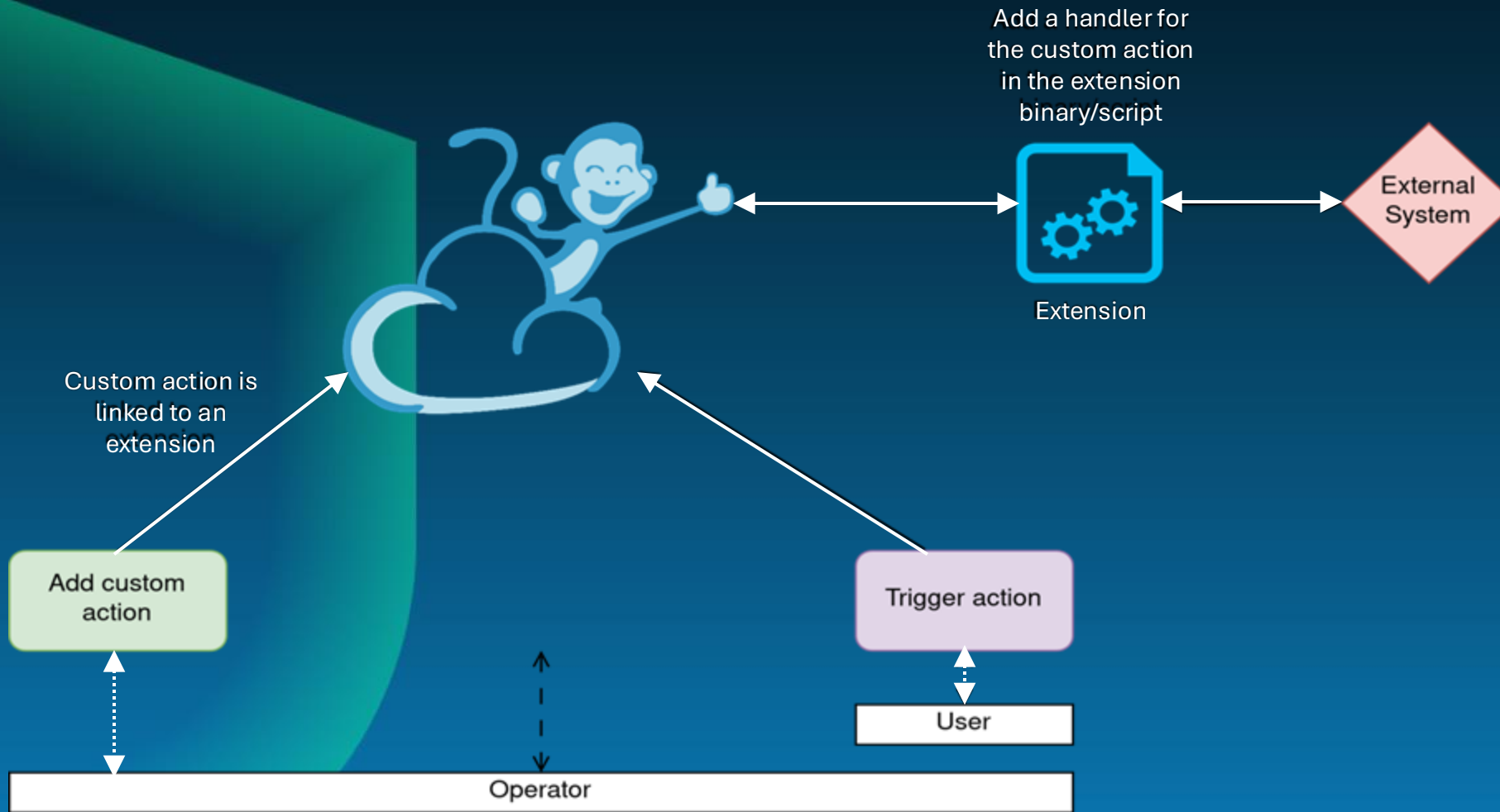


Operators can define custom actions for each extension

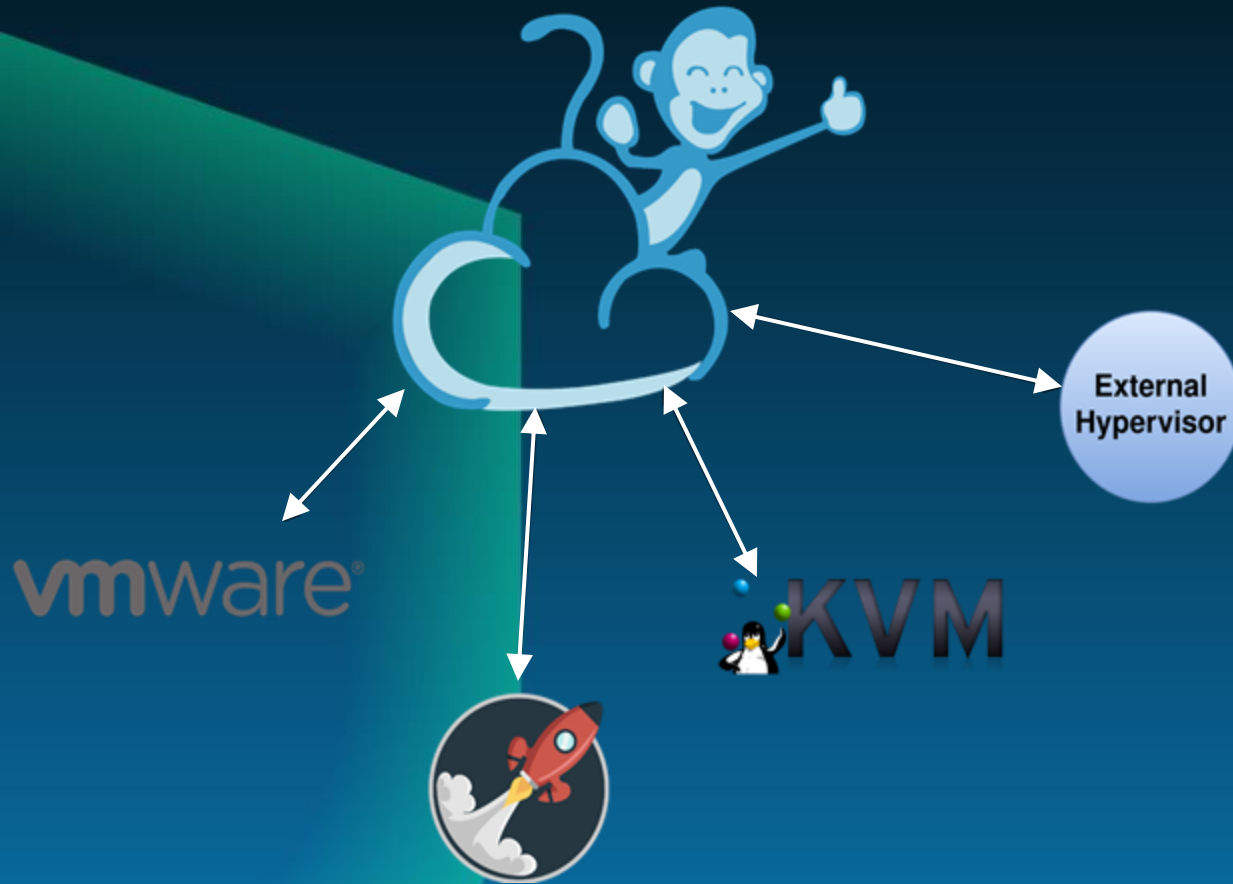
Supports user-defined input parameters, success/error messages, allowed role types

Actions can be linked to specific resource types

Custom Actions - Workflow



Orchestrator Extension



- Allows instance deployment on external systems
- Built-in extensions added for:
 - Proxmox
 - Hyper-V
 - MaaS
- Allows deploy, start, stop, reboot, expunge operations. More can be added using custom actions
- (Optional) Prepare action allows extension to update some of the fields CloudStack instance before deployment
 - Eg. MAC address for the Instance

Built-in Extensions

[Home](#) / [Extensions](#) ? [Refresh](#) [All](#) [Create Extension](#) [+](#)

Name		State	Type	Path	Availability	Created
MaaS	Inbuilt	Enabled	Orchestrator	/usr/share/cloudstack-management/extensions/MaaS/maas.py	Ready	11 Nov 2025 08:58:41
HyperV	Inbuilt	Enabled	Orchestrator	/usr/share/cloudstack-management/extensions/HyperV/hyperv.py	Ready	11 Nov 2025 08:58:40
Proxmox	Inbuilt	Enabled	Orchestrator	/usr/share/cloudstack-management/extensions/Proxmox/proxmox.sh	Ready	11 Nov 2025 08:58:40

Showing 1-4 of 4 items [1](#) 20 / page



Adding Extension

- Basic details:
 - name
 - path
 - type
 - type-specific configuration
- Optional metadata in form of key-value pair which will be passed to the binary/script

Create Extension ?

* Name ⓘ

Firecracker

Description ⓘ

Description of the extension

Path ⓘ

/usr/share/cloudstack-management/extensions/firecracker/

firecracker.py

* Type ⓘ

Orchestrator

Requires Prepare Instance ⓘ

☐


Configuration Details ⓘ

Details to be sent to the extension on any operation.

Key

Value

Add

Key	Value	Actions
<div> No Data</div>		

Enabled ⓘ

☒

Cancel

OK

Registering Extension Resources

- For Orchestrator Type:
 - Clusters
 - Hosts
 - Templates
 - Service Offering (Optional)
- Optional metadata in form of key-value pair which will be passed to the binary/script

Add Cluster ?

* Zone ⓘ

ref-trl-10083-k-Ma9-marco-sinhoreli

* Pod ⓘ

Pod1

* Cluster name ⓘ

my-extension-cluster

* Hypervisor ⓘ

External

Proxmox

Configuration Details ⓘ

Arch ⓘ

Intel/AMD 64 bits (x86_64)

Dedicated

Add Host ?

* Zone name ⓘ

ref-trl-10083-k-Ma9-marco-sinhoreli

* Pod name ⓘ

Pod1

* Cluster name ⓘ

proxmox-cluster

* Host name ⓘ

the host URL

Host tags ⓘ

list of tags to be added to the host

Configuration Details ⓘ

Details to be sent to the extension during any operation for this resource.

network_bridge

cloudbr1

Add

Key	Value	Actions
url	proxmox-host-01	
user	root@pam	
token	my-proxmox-token	
secret	792d6d6a-b0b6-4e05-ab...	
node	proxmox	

Using Extension (Trigger Action)

- No specific difference for Users
- For Orchestrator, Users will select the corresponding Template and the Instance will be deployed

3 Image

Type

Template ISO Volume Snapshot

OS image that can be used to boot Instances.

Operating System

CentOS Other **User**

Search

☒ Alpine Linux (Firecracker) **Featured** Public HVM

Override root disk size ☐

Total 1 items < 1 > 10 / page

4 Compute Offering

Search

Compute Offering	CPU	Memory
<input checked="" type="radio"/> Small Instance	1 CPU x 0.50 Ghz	512 MB
<input type="radio"/> Medium Instance	1 CPU x 1.00 Ghz	1024 MB

Total 2 items < 1 > 10 / page

Running Custom Action

- Run action show for the applicable resources
- Auto generated UI with value options, validations

The screenshot displays a web application interface for managing cloud resources. The main panel shows details for a 'Demo Instance' with attributes: i-2-34-QA, x86_64, and External. The status is 'Running'. A modal window titled 'Run Action' is open, allowing the user to execute a custom action. The modal includes a dropdown for 'Action' (set to 'TestAction'), a text area with the message 'This is a test action', and input fields for 'Name' (set to 'Demo'), 'Count' (set to 10), 'Email' (set to 'abc@xyz.com'), 'Password' (masked with dots), and 'Type' (set to 'Beta'). The modal has 'Cancel' and 'OK' buttons at the bottom.

Instances / Demo Instance Refresh

Run Action

Demo Instance

i-2-34-QA x86_64 External

Status
Running

ID
56a3200f-0c86-41d2-b03b-5c57c

OS type
QA-56a3200f-0c86-41d2-b03b-5c5

Run Action

Action
TestAction

This is a test action

Name
Demo

Count
10

Email
abc@xyz.com

Password

Type
Beta

Cancel OK

Firecracker Extension

About me

- **Marco Sinhareli**

- Committer @ Apache CloudStack Project
- 25+ years in IT and cloud infrastructure
- Technical Marketing Manager @ ShapeBlue



Proof of Concept – Use With Caution



The Firecracker extension is still experimental and has not been validated for production.



Testing in non-critical environments is strongly recommended.

About Firecracker



Lightweight VMM by AWS for serverless, edge, and container workloads



Boots microVMs in milliseconds with minimal memory overhead

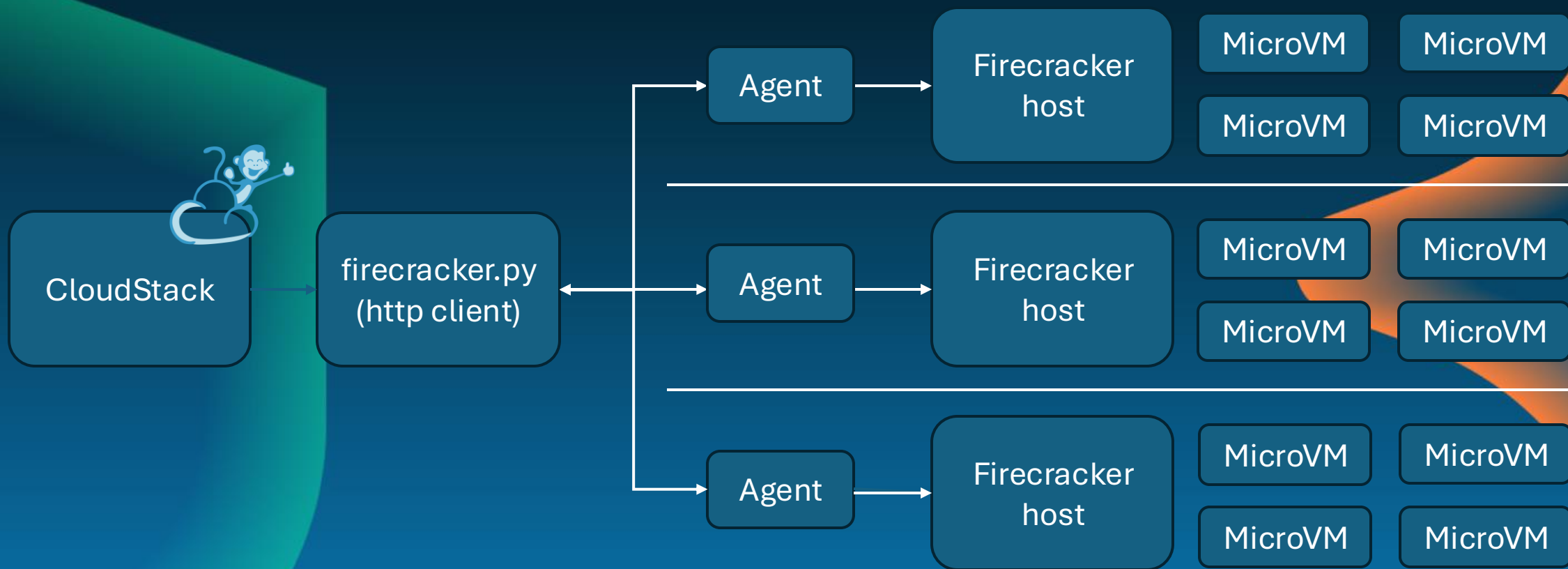


Limitation: lacks orchestration and high-level REST APIs



Goal: let CloudStack manage Firecracker microVMs as standard Instances

Firecracker Extension + Agent



Firecracker Agent Capabilities

REST API endpoints: /v1/create, /v1/stop, /v1/delete

Storage backends: file, LVM, LVM-thin

Networking backends: Linux bridge (VLAN-aware) and OVS

Process control via tmux for isolation and restart recovery

Idempotent operations with persistent state on the Host

AuthN/AuthZ via PAM

TLS + mTLS encryption for all endpoints

A large, stylized blue cloud graphic that occupies the right side of the slide. It has soft, rounded edges and a gradient of blue shades, giving it a three-dimensional appearance.

Demo

Future & What's Next

- New Types - Network, Authenticator, etc
- Usability Improvements - feedback from community
- Extension marketplace?

QA

Resources

- <https://docs.cloudstack.apache.org/en/latest/adminguide/extensions.html>
- https://docs.cloudstack.apache.org/en/latest/adminguide/extensions/inbuilt_extensions.html
- <https://github.com/msinhore/cloudstack-firecracker-extension>