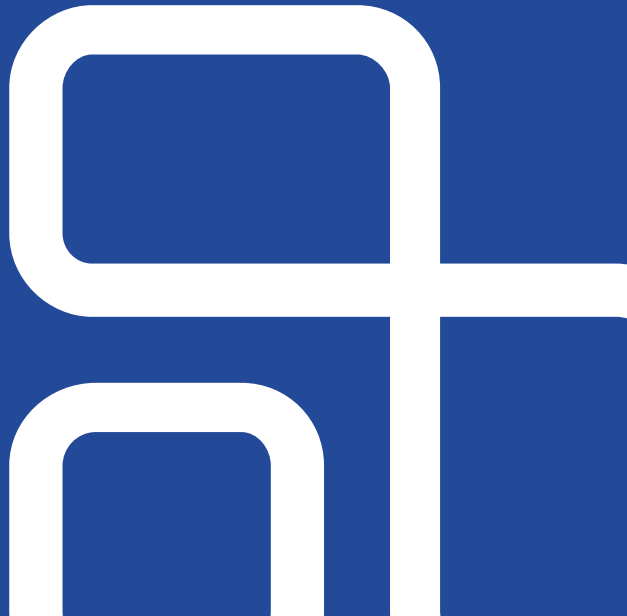
Getting Started with Anyscale

Charles Greer

Solutions Architect, Anyscale

Javier Redondo

Product Manager, Anyscale



anyscale

Agenda

1. Why Anyscale
2. Anatomy of a Ray Program
3. Scaling Ray Applications
4. Using Anyscale and the Cloud

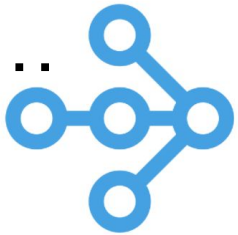
Not Agenda

1. Machine Learning
2. Ray Libraries
3. Integrations
4. Complex Dependencies

Why Anyscale?

- You've got some computational task to be done.
- One machine is not enough.

Writing for more than one machine was hard…
until Ray



Use Ray and Run it on Anyscale

- Ray makes distributed computing easy
- Ray makes Machine Learning workloads easy to scale

Ray and Anyscale make it easy compute at massive scale.

What is computing at massive scale?

- Python functions (Ray Core)
  - "I want to run OCR on 5 million documents"
  - "I want to run time series forecasting on 100k features"
- Machine Learning Tasks
  - Hyperparameter tuning
  - Distributed training
  - Model Serving
- Simulations
  - Digital Twin
  - Video Game Learning

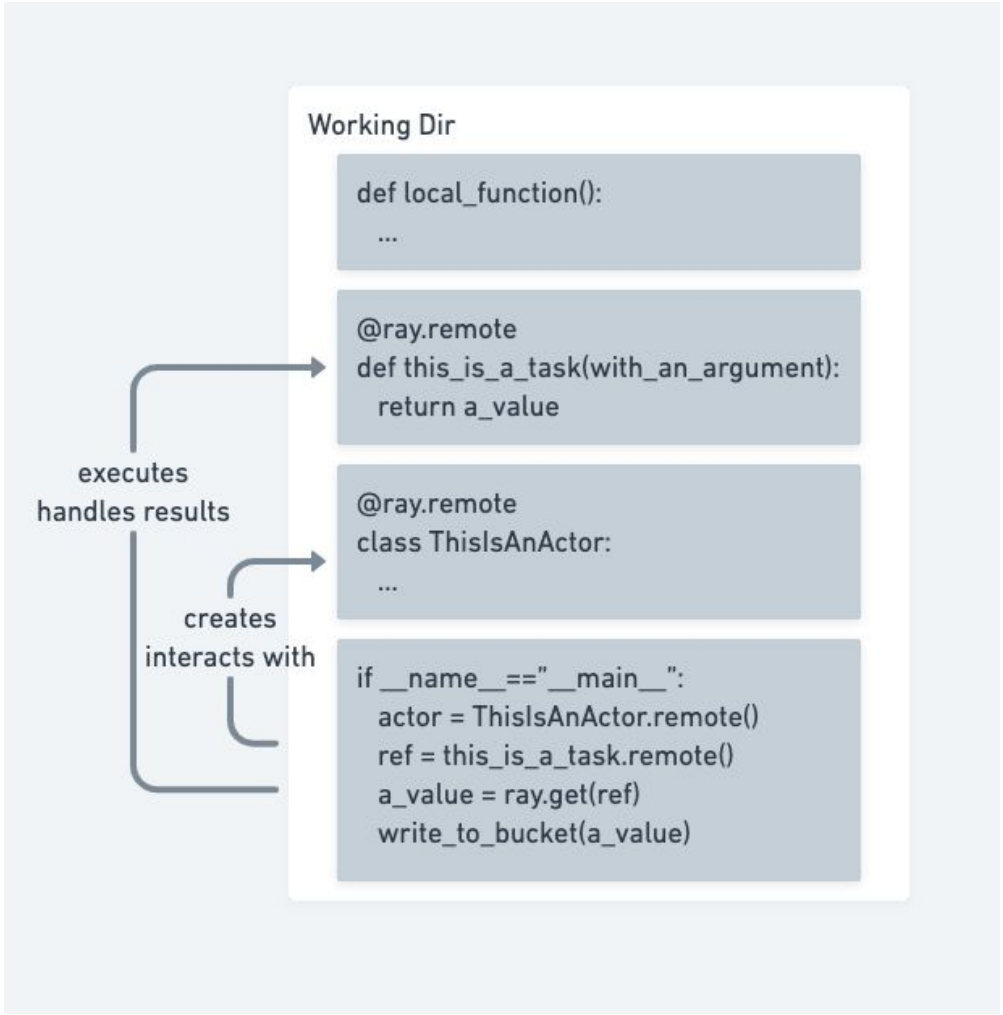
Before we get started

- Use Anyscale and Ray docs for tutorials and reference
  - https://www.ray.io/docs https://docs.anyscale.com
- Slides and code will be available on github.
  - https://github.com/anyscale/getting-started-webinar
- Anyscale has a great series of meetups, summits and webinars
  - https://www.anyscale.com/events

Pause…

… Ray Anatomy

A Picture of Ray



What is Anyscale doing? What does your code do?

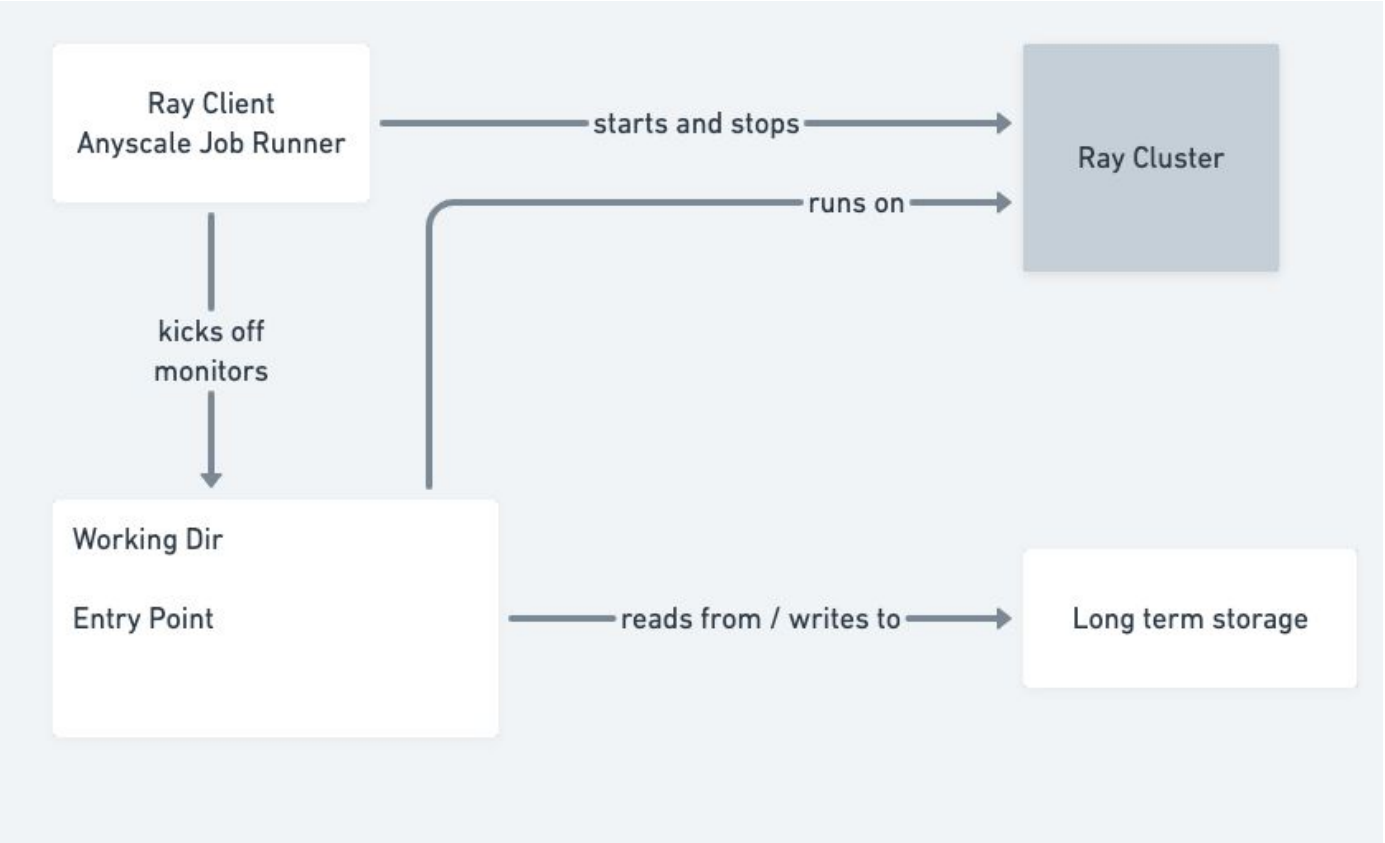
Ray and Anyscale Together

- Make copies of your code
- Manage remote tasks
- Hold function return values
- Get more machines as needed
- Get rid of unused machines
- Give you observability tools

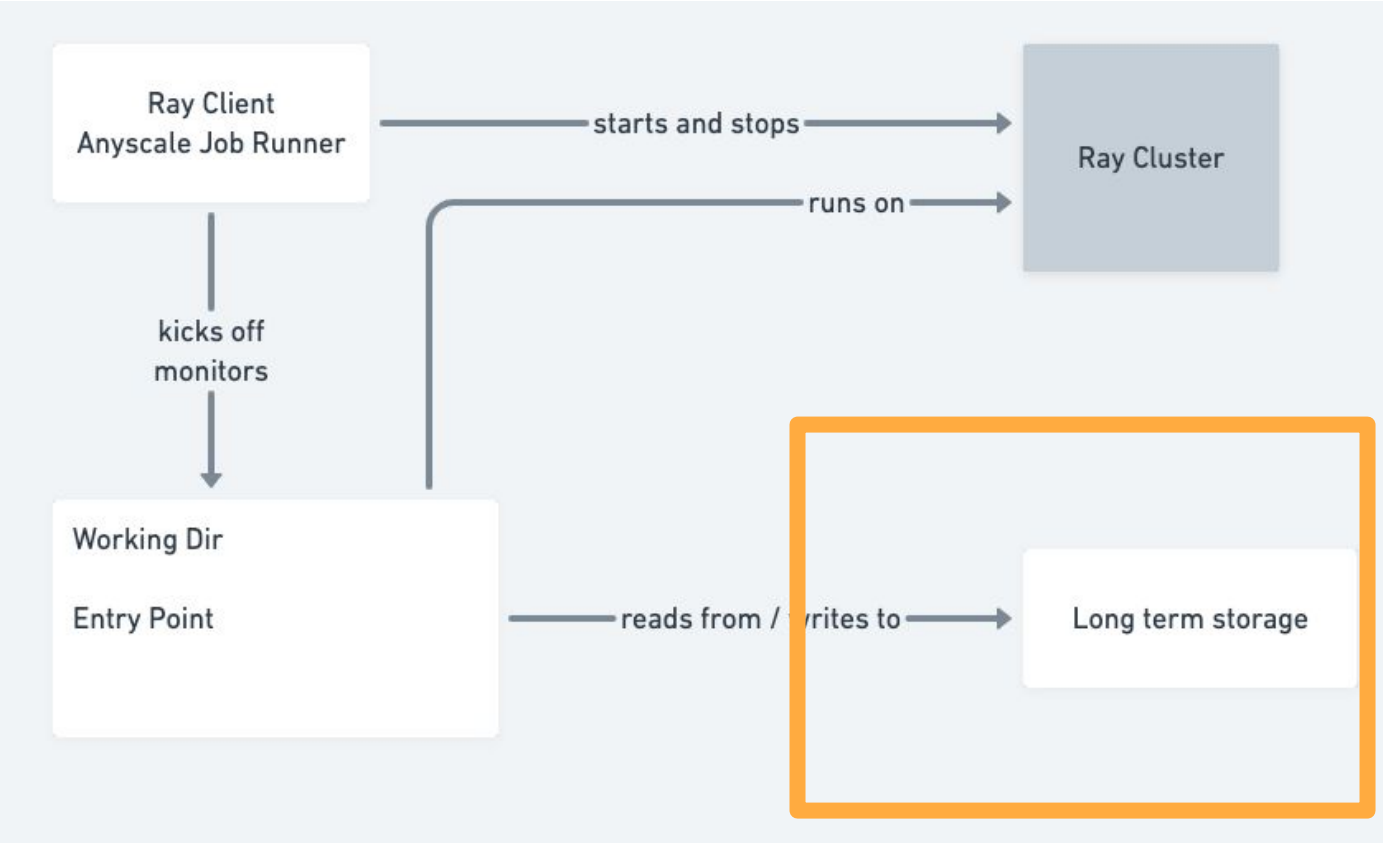
Your Code:

- Calculates things
- Writes logs
- Reads input
- Processes data
- Writes output
- Integrates with 3rd party systems
- Has Bugs

Cloud Ready Applications



Cloud Ready Applications



Pause…

… [Any]scaling

Scaling Ray

num_cpus

```
ray.init(num_cpus=100)

Actor.options(num_cpus=0.01)

@ray.remote(num_cpus=1)
```

Cluster resources

```
ray.cluster_resources()


ray.autoscaler.sdk
  .request_resources(
  num_cpus=100)
```

Run Interactively, on Anyscale

New Cluster Every Time:

- `ray.init("anyscale://")`
- `ray.init("anyscale://project/")`

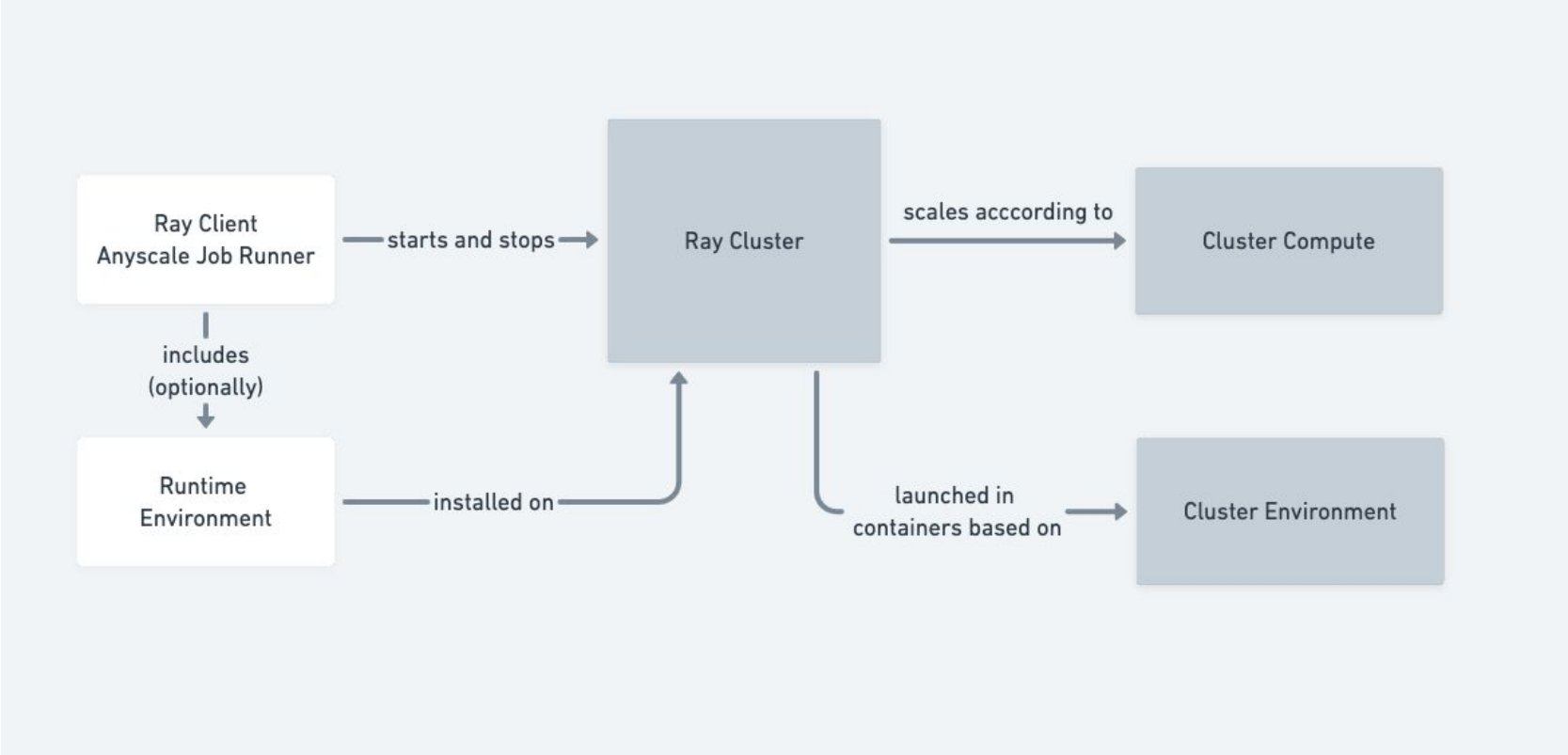
Start then re-use an existing Cluster:

- `ray.init("anyscale://cluster")`
- `ray.init("anyscale://project/cluster")`

Pause…

… Your Environment(s)

Environments on Anyscale



Cluster Environment

- **Docker Image** for your Anyscale Compute
- Build in UI or SDK
- Base image selection
- Dependencies
  - Conda
  - Pip
  - Debian
- Post-install Commands
  - Custom setups
  - Manipulate .bashrc*

Runtime Environment

- Specific to Session, Job, or Services
- Dependencies
  - Conda*
  - Pip
  - Environment variables
  - Working directory
- Installed at **launch time**

Cluster Compute Config

- Machine types
- Region
- Cloud Provider
- Autosuspend
- Create in UI or at Runtime

For all: Use by

- ray.init()
- Jobs yaml configuration
- Services yaml configuration

Using Cluster Environments and Cluster Compute Configs

```
ctx = ray.init("anyscale://getting_started/my_cluster",
               runtime_env={"working_dir" : "."},
               cluster_env="demo-with-aws:3",
               cluster_compute="demos-s3-access",
               )
```

Using Configs - S3 on Fully Managed Anyscale

- Create a role in your Amazon Acct that Anyscale can Use

  https://docs.anyscale.com/user-guide/configure/access-resources-from-cloud/overview

- Give this role permissions to access S3
- Configure a Cluster Compute to leverage this role
- (optional) Configure a Cluster Environment to install the AWS CLI
- Use them all together

Pause…

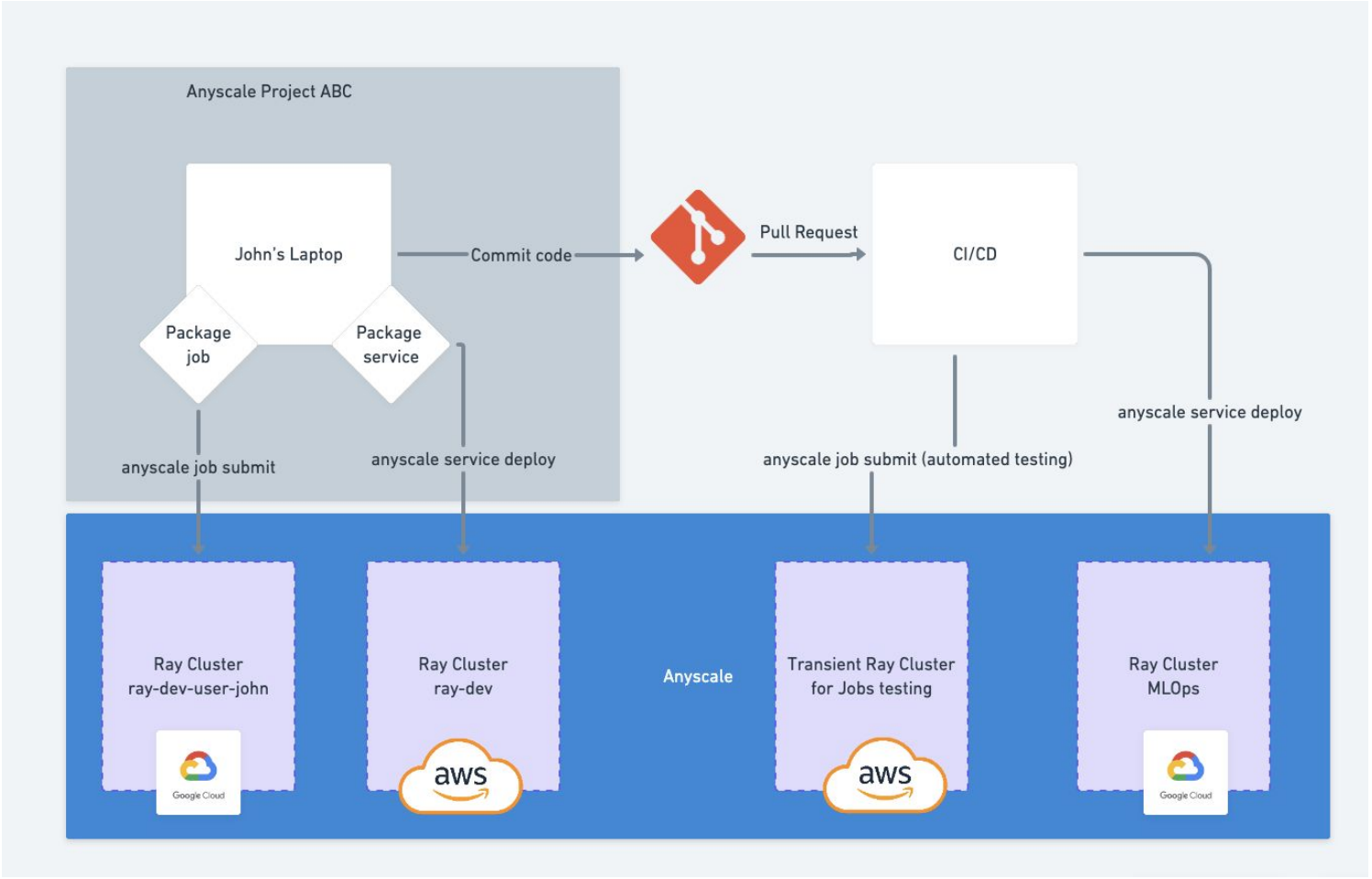
… Anyscale Jobs

Anyscale Jobs

- Your code works - ready to give to operations team.
- Hands-off production runs.
- Cluster lifecycle management by Anyscale.
- Use from Anyscale CLI or SDK

- ***For long-running processes, use Anyscale Services and Ray Serve***

Jobs and Services



Anyscale Jobs

```
anyscale job submit my_job.yaml
```

my_job:yaml:

```
name: webinar-job
cloud: anyscale-managed-2
Runtime_env:
  working_dir:
"https://github.com/anyscale/getting-started-webinar/archive/refs/heads/master.zip"
entrypoint: "python a_script.py 1012"
max_retries: 3
```

Pause…

The Adventure Continues

- Ray Libraries for ML
  - Ray Train
  - Ray Tune
  - RLlib
  - Ray Serve
- Anyscale Services
  - Scaling Model Serving
- Your use case……

Supercharge your Ray journey on Anyscale

Accelerate time to market

Fully-managed service
Focus on innovation; not infra ops

From the creators of Ray
Access to Ray experts

Built for dev -> prod journey
Scale from laptop to cloud seamless;
Easy CI/CD integration

Enterprise ready

Observability
Get full visibility into your Ray workloads

Multi-Cloud
Diversify and deploy your workloads across public clouds with a click of a button.

Simplify your MLOps with Anyscale

Effortlessly deploy AI workflows and models into production with your existing CI/CD tools.

Production jobs & services
Deploy ML workflows & models into production with ease

Observability
Monitor health with event logs and prebuilt dashboards

App packaging
Package apps, incl. all code and library dependencies

APIs & SDKs
Automate and integrate into your workflows (eg. CI/CD)