cnvrg.io Metacloud

Frequently Asked Questions

Q. What are cnvrg.io and Metacloud, and what do they do?

cnvrg.io is a **self-hosted** software platform to simplify and automate the continuous training and deployment of AI and ML models. It manages the entire lifecycle from data preprocessing, experimentation, training, testing, versioning, deployment, to monitoring.

Metacloud is the cnvrg.io platform offered as a **managed service** -- there's nothing to install.

Throughout this document, we'll refer to Metacloud and cnvrg.io collectively as "the Platform."

The biggest challenge today facing AI and machine learning at scale is that data scientists are spending less time on the data science work they'd rather be doing, and too much time on unproductive tasks like configuring hardware (GPUs, CPUs, accelerators, and storage), standing up cloud computing resources, configuring containers and container orchestration software like Kubernetes, or managing complex heterogeneous environments like hybrid cloud infrastructures. The Platform makes it easy to automate all of this work and manage end-to-end AI and ML pipelines across all environments, whether on premises, in the cloud, or in hybrid environments.

Also, developers need the ability to choose the best of breed compute and storage solution for each workload, based on each architecture's cost and performance trade-offs without the overhead of standing up completely new stacks and negotiating commercial terms for each, which might take months to set up.

The Platform gives you point-and-click ease for choosing your own compute and storage resources, from partner-provided options from market leaders like Intel, NVIDIA, Dell, and many others, or from the major cloud providers. You can easily place any workload on the optimal resource, whether that resource is on-premises or in the cloud.

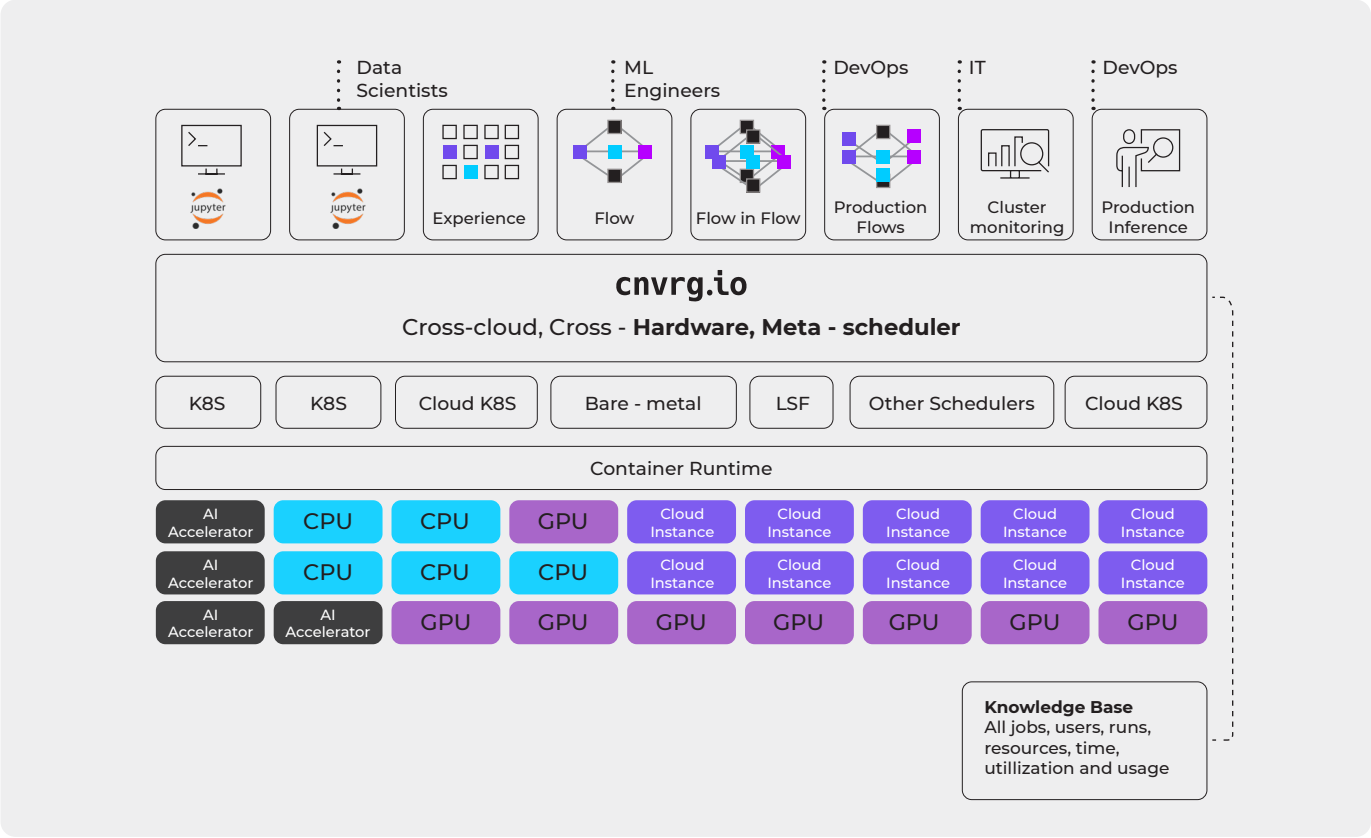

Figure 1: cnvrg.io Platform

How does it work?

Broadly speaking, there are two main layers:

Application layer: Manages the back-end and front-end services, including the database, object storage, metadata services, and more.

Compute (workers) layer: This is where the machine learning jobs and workload are executed (experiments, model training, and so on).

At a more granular level, we provide several capabilities for data scientists:

Control plane: the Platform provides a drag-and-drop graphical interface for constructing and managing end to end AI and ML pipelines, from training to production, and dashboards for monitoring the state of jobs and resources.

Self hosted cnvrg.io requires users to deploy the control plane to the runtime environment as a purpose-built Kubernetes Operator via Helm charts. See "Orchestrator and Workload Scheduler" below for more on our use of Kubernetes. But whether self-hosted (cnvrg.io) or managed (Metacloud), the functionality provided by the control plane includes:

- Notebooks and workspaces: Interactive environments for developing and running code, and for collaborating with the rest of your team. The environment is pre-configured -- all dependencies are preinstalled. All the files and data in the workspace will be preserved for you, across restarts. A workspace has automatic version control and scalable compute available, so that you can use unlimited compute resources to do your data science research. cnvrg.io has built-in support for popular notebooks like JupyterLab, JupyterLab on Spark, R Studio, and Visual Studio Code.
- AI and ML libraries: Building ML pipelines and continual learning are key to an effective machine learning workflow. Reproducible and modular code components are core components of any such workflow. The AI Library in cnvrg.io facilitates these goals. It is a specially-built package manager for ML components designed specifically for machine learning. It helps data scientists and developers build machine learning components and reuse them across projects.

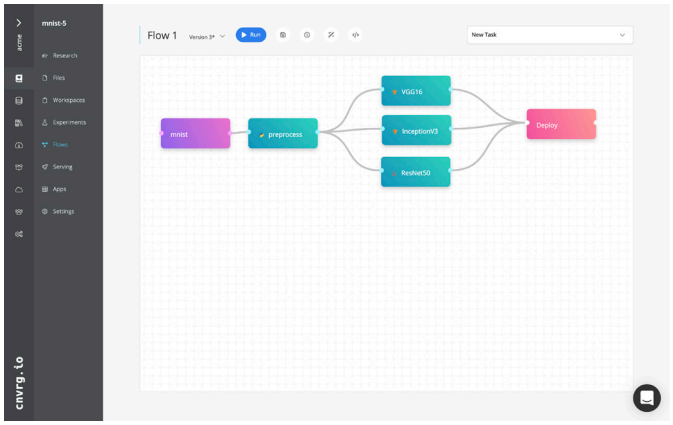

Figure 2: Drag and drop ML pipelines with cnvrg.io

The Platform provides a visual workflow interface for designing end-to-end ML pipelines. The visual environment improves the reusability and traceability of ML components as well as its optimization for different environments. cnvrg.io.io ML pipelines capabilities include automatic hyperparameter tuning as well as out-of-the-box integration with runtimes such as Spark or Kubernetes. The Platform also includes performance monitoring retraining triggers based on the runtime behavior of ML workflows. cnvrg.io comes pre-configured with many data preprocessing, AI and ML library components that you can easily invoke from the UX with point-and-click simplicity. You can also easily create and register your own algorithms -- simply indicate the Git repository where your code resides, and cnvrg.io will import your code and create a new branch.

Dataset management and version control: the Platform enables native integration with data sources such as Snowflake, S3, relational databases and many others. The Platform includes a labeling interface that facilitates the creation of training datasets. Additionally, you can associate training datasets with models creating the necessary feedback loops for optimization and retraining.

The Platform manages your dataset with an internal version controlled system, automatically, so you can track your dataset at every stage. Any action is written as a new commit, so that you are able to browse and revert to specific versions. Versioning gives you the confidence to use your dataset as you need -- cnvrg.io will always keep it safe and controlled.

The Platform also has native integration with Git. Developers can use an external Git repository to track code and files, while using cnvrg.io to track data science resources (models, experiments and more).

- Data caching layer reduces the need for costly data egress to different environments, leaving training data co-located with compute resources wherever they are. Caching also accelerates training and re-training.

Orchestrator and workload scheduler: the Platform makes use of Kubernetes as an orchestration, scheduling, and scaling layer. It makes jobs portable across environments and scales resources up and down on demand. We are able to use Kubernetes' own native mechanisms, such as taints and tolerations, to place workloads only on appropriate nodes. For example, we can ensure that jobs requiring GPUs for training don't land on nodes with only CPUs. cnvrg have also created a Metascheduler to address some of the limitations of native Kubernetes, which lacks a holistic approach to multi-cluster scheduling. With the Metascheduler, cnvrg is able to intelligently schedule pods across clusters -- whether on premises or in the cloud -- to alleviate unwanted conditions like resource contention and deadlocks.

Heterogeneous compute and storage: Even in a single pipeline, developers may need different types of compute and storage resources at different stages. The Platform abstracts compute and storage into a utility by making it seamless to connect a variety of compute and storage resources -- CPUs, GPUs, specialized AI accelerators and the like -- whether they are the cloud, on-premises or at the edge, and to consume them in a cloud-native way.

For example, a developer could preprocess data on their own Xeon CPUs, train on NVIDIA GPU compute instances on a Dell EMC rack, and then serve on CPUs hosted on AWS or Azure, depending on the optimal combination of performance and cost. All of this happens with point-and-click ease.

The Platform also provides developers with the convenience of a marketplace for compute and storage resources. Developers can select these from a menu of containerized OEM and partner-provided options and consume as if they were any other cloud-native resource.

Monitoring and reporting. The Platform provides a wealth of information on jobs, resources, health, and consumption, including:

- **Current active jobs**
- **Current pending jobs (queued or initializing)**
- **Amount of compute resources**
- **Assigned and total available GPU cores**
- **Assigned and total available CPU cores**
- **Assigned and total memory**
- **Assigned and total storage**

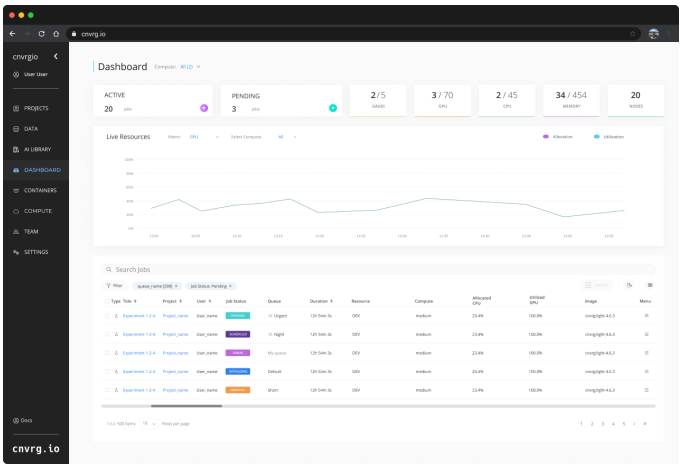

Figure 3: Monitoring dashboard in Metacloud

Q. What are the main use cases?

Our Platform can be used for a wide variety of AI and ML use cases, including autonomous driving applications, recommendation engines, classifiers, clustering, natural language processing, computer vision, and many more types of applications. We provide an extensive off-the-shelf library of popular algorithms, and data scientists can easily extend the use cases that they can address with their own algorithms simply by linking their Github repos to the Platform.

Q. How can I deploy and consume the Platform? How is it priced?

There are three main offerings:

Convrg.io CORE is our free community data science Platform. It can be installed on-premises or in a cloud environment. It's limited to 4 users, 8 CPUs, and 16GB of memory. Review the docs, ensure that you have all of the dependencies installed, and then install the software with the provided Helm charts. cnvrg.io.io support is not part of this offering.

cnvrg.io is a self-hosted platform, with identical capabilities as CORE but with the ability to use resources from multiple cloud providers, no limitations on CPU, memory, or storage you can add, a dedicated customer success engineer, and 24/7 support. It requires an enterprise license, please contact us for specifics of pricing.

Metacloud is a managed version of cnvrg.io that allows data scientists to quickly develop, train, deploy and manage models across any infrastructure, but with the ability to keep your data and workloads in your own organization network. Our Metacloud announcement blog has more details. Metacloud is available as:

- As a free version (limited to 8 CPUs and 16 GB, but with unlimited usage). Support is the same as CORE above.
- On a pay-as-you-go model, where you can connect and manage unlimited resources and pay only for the resources you actually use. Includes 24/7 customer support.
- With an enterprise license, that also includes a dedicated customer success engineer.
- Contact us for more information on pricing.

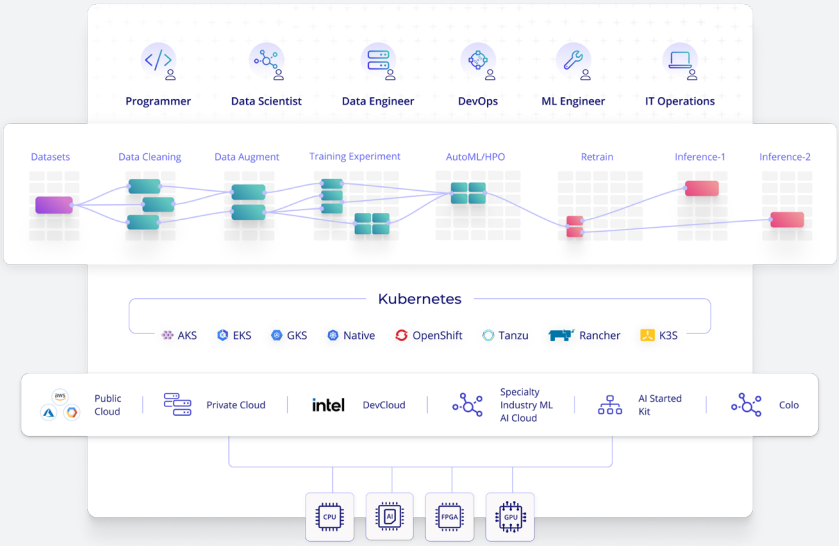

Figure 4: Metacloud

Q. Which Kubernentes distributions are supported?

We are compatible with Kubernetes distributions from AWS, Azure, and GCP, as well as any other CNCF-compliant stack, including OpenShift (catalog) and Rancher.

Q. If I consume hardware resources from another provider, how is the pricing and billing for those services handled?

You are simply billed by your cloud provider for the rates that you agreed upon. If you are connecting your own on premises resources (e.g., a GPU cluster), then there is no charge for that.

Q. What resource providers does the Platform work with, and why is the idea of compute and storage as a utility so important?

Our goal is to make the lives of data scientists as easy as possible, and part of that is making the consumption of hardware resources as frictionless as possible. Assembling a hardware bill of materials is tedious and time consuming, with lots of negotiation with different vendors sales' teams. We work with industry leading providers of compute and storage to give developers and data scientists broad flexibility to select the best resource that's appropriate for their workloads.

Q. What controls does the Platform have for data governance?

We provide several levels of access control that you can set for users in your organization. This allows you to limit what your users can see and what read/write permissions they have.

There are controls to limit who has access to what data, what data can be on what type of cluster, and what compute or storage is allowed, and what regions can be used. See our separate document on security controls.

Q. Being able to deploy workloads in heterogeneous environments is great, but how do I avoid unwanted data movement?

Data sets that are being used frequently in any environment are automatically cached. So whether workloads are running on-premises, or in the cloud, you don't need to perform an ETL operation and wait hours until your data is ready. It's already co-located with compute. And if the Platform identifies that you haven't been using this data set for a while (e.g., for a few days), then it automatically clears the cache.

Q. There are a lot of deployment options across clouds and data centers. Where can I cache data to avoid unwanted data movement?

You can create a data cache with an NFS server in a Kubernetes cluster in the environment of your choice. Setting up and connecting to an NFS server is covered more completely in the documentation.

For example, if you are storing data on S3, you can cache data on an EKS Kubernetes cluster, so you don't need to move data from S3 to another environment.

Q. Does self-hosted cnvrg.io do away with the need for devops?

No, because devops will still want to monitor resource usage and health enterprise-wide, not just AI and ML workloads. There's a need to make sure that resources are being used in accordance with policies,that resources are being used fairly, and that there are no runaway processes causing problems for other jobs running on the system. Instead, cnvrg.io helps devops with health, usage, and performance dashboards that helps them to monitor and manage more jobs without adding to their workload.

Q. Explain the role of Kubernetes in cnvrg.io. Is Kubernetes essentially invisible to me, or do I still think about networking, security considerations, scaling, cluster size, and so on?

Kubernetes is the container orchestration system for cnvrg.io. Kubernetes provides a framework to run distributed systems resiliently. It takes care of scaling and failover, manages deployment patterns, and much more.

With Metacloud, which is a managed service, we insulate data scientists and devops from the lower level configuration and operational details of Kubernetes. Just sign up, get your access credentials, and start coding.

For self-hosted cnvrg.io, you will first need to stand up a CNCF-certified Kubernetes environment. You will then need to install the cnvrg.io Operator after ensuring that all necessary dependencies are met. Please see the documentation for a more complete description of the installation process.

Q. Where can I get the software, and how can I learn more?

We have many educational resources at your disposal. Start with our documentation. Watch our YouTube videos with demonstrations, use cases, and best practices. Contact us to get a demonstration or request Metacloud access, or just download the CORE community edition to try it on your own hardware.

About cnvrg.io

cnvrg.io is an AI OS, transforming the way enterprises manage, scale and accelerate AI and data science development from research to production. The code-first platform is built by data scientists, for data scientists and offers unrivaled flexibility to run on-premise or cloud. From advanced MLOps to continual learning, cnvrg.io brings top of the line technology to data science teams so they can spend less time on DevOps and focus instead on training and deploying models into service. Since using cnvrg.io, teams across industries have gotten more models to production resulting in increased business value. For more information, visit https://cnvrg.io/.

Contact Us:
https://cnvrg.io
sales@cnvrg.io

Schedule a demo:
https://cnvrg.io/demo/