

Audio file

Your Recording 36.wav

Transcript

00:00:04 Speaker 2

Listen, we're going to look at Sagemaker domains and the Sagemaker Studio user interface.

00:00:12 Speaker 2

We've seen how to create Jupyter notebook instances. Unfortunately they do have some limitations, hence they were replaced with the newer Sagemaker Studio.

00:00:24 Speaker 2

So we're going to look in this lesson as to what those limitations were and how Sagemaker Studio addresses them.

00:00:33 Speaker 2

We're also going to look at the statutes you require to perform in order to consume Sagemaker Studio, specifically the creation of a Sagemaker domain. We are then going to look at some of the benefits and the improvements available within the Sagemaker Studio user interface. So let's think first of all about the Jupyter instances. We can create a Jupyter Notebook instance as we saw in the last demonstration.

00:01:02 Speaker 2

But those Jupiter Lab instances are created on a one to one basis, aligned to a specific user. Now if we imagine a team of multiple data scientists, each one launching A Jupyter notebook instance, we end up with lots of notebook instances. Each one is being billed against our account, and it seems a little bit wasteful to have multiple separate.

00:01:32 Speaker 2

Instances for that. What makes this worse is that these notebook instances do not shut down by themselves. They need to be manually shut down.

00:01:41 Speaker 2

So that makes it the responsibility of the user. But the user is simply focused on consuming the Jupyter interface. They maybe are not thinking about shutting down notebooks.

00:01:54 Speaker 2

Another issue we have with Jupyter lab is we don't have any experiment tracking.

00:02:00 Speaker 2

We'll talk more about that in a moment, but there's no native experiment tracking further. There's no automated git setup. It's up to the user to use the Git plugin to Jupyterlab to explicitly clone in any repository.

00:02:16 Speaker 2

Each user is responsible for managing their own instance, stopping it and starting it, and ensuring it stays stopped to eliminate unwanted spend. So if we have that increased responsibility in the user as an increased risk of extra charges. Further, we are limited in Jupyter lab to just the functionality of Jupyter lab if we want to use.

00:02:44 Speaker 2

Additional functionality that Sagemaker might expose.

00:02:47 Speaker 2

Then that's not directly available to us within the Jupyter lab interface. It'd be nicer to have a purpose built machine learning integrated development environment that would give me access to those tools as well as Jupyter. Now we mentioned experiment tracking there when we have a data scientist who's performing multiple training jobs in order to find the best model, the best combination of.

00:03:17 Speaker 2

Set an algorithm and feature engineering and hyperparameters to get the best results. Well, if we were to challenge them. Have you got the hyperparameters that you use from your best attempt 3 iterations ago? The result you might get from the data scientists could be varied, which you might get back is not sure. I'm now on my next iteration and my one after that. I've already forgotten what I did an hour ago. Now I'm not advocating that approach.

00:03:45 Speaker 2

I'm simply saying that a number of data scientists.

00:03:48 Speaker 2

Might work in a sort of continuous improvement way, but maybe are not that great at reproducibility about being able to go back to a point in time and say actually what I had earlier was better and knowing how to reproduce exactly what they did earlier because they logged precisely the combination of data sets and algorithm and features engineered at that point. So we need to make it easy for them and the way that we do that is offering.

00:04:18 Speaker 2

A framework.

00:04:19 Speaker 2

So the data scientist doesn't need to be systematic in their approach to experiment management, but we need to make it easy for them to record their inputs and outputs as an experiment like

experiment one is using this data set, this version of the algorithm, and this particular training job with these hyperparameter values. Then when they look at the resulting model and look at the evaluating its performance, they can compare and contrast the results.

00:04:48 Speaker 2

Against each different training iteration. Now natively when we had notebook instances in Sagemaker, there wasn't native support to do that then data scientists would need to use a third party tool such as Comma ML or True Era, something like that. And that would mean an additional subscription cost to that software supplier. So wouldn't it be nice is as we are solving some of the limitations of Jupyter Notebook?

00:05:19 Speaker 2

If we could solve that problem too. So the solution that AWS have is the Sagemaker Studio built upon the Sagemaker domain. Let's take a look.

00:05:31 Speaker 2

A Sagemaker domain is an administrative boundary we create within Sagemaker. We can have multiple Sagemaker domains if we wish, but a Sagemaker domain has a number of component parts. Firstly, it has a common file system. This file system is based on the EFS file system, the Elastic File System. Now this is essentially A serverless managed files share based on the NFS protocol. It's been available within the.

00:06:01 Speaker 2

To service for many years, but here we automatically create you file share that can be mounted by NFS.

00:06:09 Speaker 2

Then we need a user profile. A user profile is made within the domain. I can't use the Sagemaker domain unless I have a user profile. Having an AWS user account is not enough here. I need a user account to get into the Sagemaker console. But if I want to launch into Sagemaker Studio, I need to have a defined user profile for the domain I wish to consume.

00:06:40 Speaker 2

Then within the Sagemaker domain, I would launch the Sagemaker Studio. This would open up as another tab in my browser. Now when I get into the Sagemaker Studio, I then have a range of options of different applications I could use. The primary one I'm going to be interested in using is Jupyter Lab.

00:07:02 Speaker 2

It's the one we are used to, so I can now get into Jupiter Lab via the Studio interface. But the Studio interface gives me so much more than just Jupyterlab. I can consume other services directly from that interface, whether it be to create large compute clusters, to run training and large language models, or whether to access capabilities like Jumpstart, which will allow me to immediately deploy marketplace models.

00:07:33 Speaker 2

There's numerous features accessible to me directly within Studio, not just playing Jupiter Lab, but we have other applications accessible to us as well, like Rstudio. Many data scientists and statisticians have used the R programming language in the Rstudio integrated development environment and are happy to continue using that for their machine learning development, so we provide that as a hosted user interface.

00:08:01 Speaker 2

As well. So you are giving the option to the user profile. Which application would you like to use? And then I click on that option and a new interface is opened up. So we can have hosted Jupyter or hosted Rstudio. Although you will need a separate license from R in order to use Rstudio.

00:08:21 Speaker 2

We also have Code Editor, which is essentially Microsoft Visual Studio Code hosted within Sagemaker Studio. OK, that gives us an alternative interface to Jupyter. And lastly, we have Studio Classic. When Sagemaker Studio first launched, it looked a little bit different and that was what we originally called Studio, but we've now moved that to be called Studio Classic.

00:08:49 Speaker 2

And we're really just leaving it in there to gain access to some legacy features that haven't been turned off yet. If you're brand new to using Sagemaker Domain and Studio, ignore Studio Classic. It's really only there for people who started with the earlier version of Stage Maker Studio.

00:09:10 Speaker 2

We also need to specify an execution role. In other words, what security context are we going to run the studio under, which will determine what features we are able to use of Sagemaker and will determine what, for example, S3 buckets we can consume or indeed interface with any other AWS service.

00:09:32 Speaker 2

So a Sagemaker domain and the Sagemaker Studio is going to give us a number of benefits. It's going to give us access to shared spaces. In other words, I can have multiple users of my Sagemaker domain who can collaborate in the same shared space. I have the option of sharing Jupyter notebooks via links to make it, again, easier to collaborate.

00:09:57 Speaker 2

I can have simple integration with git whereby my Jupyterlab interface will be automatically launched with my git repository preloaded into it, encouraging the use of git version tracking right from the beginning.

00:10:14 Speaker 2

I get access to experiment tracking now, experiment tracking. We have something called Sagemaker Experiments, and yet again with AWS, everything is changing all of the time and we're

seeing a gradual shift away from Sagemaker Experiments to something called ML Flow. But I think the point still stands that by moving to Sagemaker Domain and Sagemaker Studio, there are options for us now about using.

00:10:44 Speaker 2

And native experimentation management solution, whether it's Sagemaker experiments or the newer ML flow, we gain access to automated workflows within Sagemaker Studio. We can create and manage Sagemaker pipelines. Now a pipeline is just an orchestrated set of activities in a sequence. So simplest level, I might have two pipelines, one for training a model and one for.

00:11:14 Speaker 2

Inference A.

00:11:16 Speaker 2

Training pipeline might for example, clean up my data, feature, engineer it and then train the model. An inference pipeline might take input from a user, clean and prepare it, pass it to the model and get a prediction back. So it allows us to build automation around what we are trying to do. So Sagemaker pipelines is a major advantage to have and we only get it if we use Sagemaker Studio.

00:11:45 Speaker 2

We further get fine grained access in terms of IAM policies, so we can be absolutely explicit about what permissions we want to delegate and for what purposes within our ML pipeline. Maybe to specific datasets or maybe to certain algorithms, or maybe to certain Sagemaker pipelines. All of these can be locked down so that you can only consume the ones that you've explicitly been granted for.

00:12:15 Speaker 2

So we have a Sagemaker domain which is an administrative boundary for Sagemaker Studio. We are permitted to create more than one Sagemaker domain. If you need to ring fence maybe different projects or in different business units away from one another. We need to define our user profiles within each Sagemaker domain in order to connect to it and consume the Sagemaker Studio interface.

00:12:42 Speaker 2

Each Sagemaker domain has its own Elastic File system, In other words, a shared managed file system that can be accessed by all users of that domain. Now a small caveat here, EFS, although part of a Sagemaker domain, played a far greater role in Sagemaker Classic now that Sagemaker Classic is going away.

00:13:08 Speaker 2

We are kind of de emphasizing using EFS.

00:13:12 Speaker 2

Because of performance reasons, and I'll talk more about that later on.

00:13:17 Speaker 2

Let's now look at how we can gain access to the Sagemaker Studio. Here we are in the AWS Management Console.

00:13:25 Speaker 2

I've already clicked into the Amazon Sagemaker AI link and we are now looking at the Applications and Ides section of the sidebar on the left hand side. Now if we want to get into this studio, we're going to have to click the Studio link. But notice just before we do this that we have a warning in this particular screenshot. This is just telling us that Jupyter Lab 3 notebooks on Sagemaker Studio Classic is going end of support.

00:13:55 Speaker 2

On the 31st of December. Now we're well past that point now, however, there are still customers out there who are using Sagemaker Studio Classic even though we're past the point of end of support. So just be aware of that. Ideally, we should all be in Jupiter Lab for at least and we should be using Sagemaker Studio here. We're going to click on the link to create a Sagemaker domain. The domain is the prerequisite for gaining access to Sagemaker.

00:14:26 Speaker 2

When we create our domain, we'll be prompted if we want a single user quick setup or if we want to set up for organizations.

00:14:35 Speaker 2

So which one should we choose?

00:14:38 Speaker 2

Well, if we are going to create a Sagemaker domain purely for testing or just learning Sagemaker Studio, then the quickest time to value would be with a quick setup. But there are some limitations.

00:14:55 Speaker 2

This will use your region's default virtual private cloud. Now that is network, which is intended for experimentation and has no real network security rules set up, so it's not the best environment for production workloads. It's only really there to allow you to experiment. So we would say if you use quick setup it should only be used for personal development or proof of concept and nothing that is production.

00:15:26 Speaker 2

So when would I use the manual setup? Well if I want total control over security, infrastructure and compliance. In other words anything pre production or production you want that to be a manually set up Sagemaker domain. For example maybe you want to use encryption and custom encryption

using keys that are defined and allowed for use for that purpose. Or maybe you want to leverage customer authentication.

00:15:56 Speaker 2

Mechanisms. Maybe you are already using the AWS Identity Center for Federated login using Microsoft Entra. Or maybe you need to have very fine grained control over which users or which security context is used for running a training job. Anything like that definitely requires a manual setup.

00:16:19 Speaker 2

So we should be choosing manual setup when we need that precision over security, when we need integration with existing infrastructure, which could simply mean an existing network, virtual private cloud, a VPC where you have existing resources that you need to interact with rather than using the default VPC.

00:16:42 Speaker 2

When you need strong compliance with organizational policies, so we need to integrate with those policies set at an Oracle level. When we need to have cost optimization for our production deployments, We have control over every element of what we provision and whenever we need any custom network provision. For example, maybe your consumption of Sagemaker comes from a private network.

00:17:08 Speaker 2

Maybe an on premises network that is rooted in via direct connect? Maybe we don't want to support Sagemaker accessed over the Internet? Any rules like that? You're going to need a manual setup of your Sagemaker domain.

00:17:23 Speaker 2

And as I mentioned, custom authentication methods. So leveraging Federated login, Microsoft Entrust through Identity Center, anything that is SAML based is good to have to be done through a manual setup of your domain.

00:17:38 Speaker 2

Once your Sagemaker domain has been provisioned, you will see the interface changed slightly. Rather than being prompted to create a Sagemaker domain, you are now being prompted to open Studio when you are in the Studio section of the Applications and Ides Navbar item notice.

00:17:59 Speaker 2

Just above where it says Open Studio, you've got user profile. Now in this particular example I used quick setup, so it's created me as Sagemaker domain and it's created me a default user profile. So I can now pick what user in that Sagemaker domain I want to use. And I could click the Open Studio link If however, I wanted to look at other users that are available for me to use in that domain or if I want to create a new.

00:18:29 Speaker 2

User profile I would have to come across on the left side of the NAV bar, find admin configurations and find domains. If we click on domains we should see the Sagemaker domain that has been created for us. Once we've clicked on domains we can see our Sagemaker domain that has been created. In this example I can see mine is called Quick setup domain and then a unique date and timestamp after it. But if you were creating a manual domain you would have had the ability.

00:18:59 Speaker 2

To influence that name, I could see that my domain is in service and ready to go.

00:19:04 Speaker 2

If I want to see the properties of that domain, including what user profiles have been defined within it, I would need to click directly on that blue text where it says Quick Setup domain. In other words, the name of the domain itself is a link.

00:19:21 Speaker 2

So we can now see our Sagemaker domain properties. We can see more about when it was created or last modified, what virtual private cloud that it's using, and most importantly that it is ready for use. Now I can see a number of tabs along the top. We are in the Domain settings tab by default.

00:19:40 Speaker 2

I want to go to the User profiles tab and see what users are defined in this domain that will be allowed to launch the Sagemaker Studio.

00:19:50 Speaker 2

So here we are in the User profiles tab of the Sagemaker domain. I can see in this example I have a single user profile defined. Now user profiles do equate to a person. So myself as Alistair I would have a login. Here I would be defined as a user profile. If I wanted another user to also login then I would need to add that user profile here to enable them to be permitted.

00:20:20 Speaker 2

To use one or more of the Siege Maker Studio applications. Now let's say I want to launch Age Maker Studio. I can see for each user profile listed there's a button on the far right hand side. That button simply says launch, but it's a button and a drop down list at the same time. So if I click on the word launch, what else see there?

00:20:46 Speaker 2

Is what apps can I access Now most of the time the app that you will be going to access will be the Sagemaker Studio, but there are other options there such as Spaces, Profiler, Tensorboard, and Canvas and we'll be talking more about them later on for now, this is our way in to the Sagemaker Studio. Now following clicking that link, what will happen is I will get a new browser window opening up and that new browser window will.

00:21:15 Speaker 2

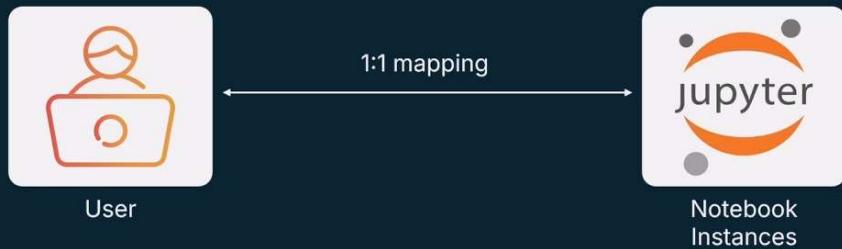
Will be pointing at a URL that will give me Sagemaker Studio. You can see in the URL that I'll have Studio and then I'll have a unique identifier and then I'll have the region dot sagemaker dot AWS. So this is a different URL from where the management console is being delivered to you always at studio dot region dot sagemaker dot AWS.

The screenshot shows a dark-themed video player interface. At the top center is the KodeKloud logo, which consists of a blue cloud icon with two white arrows pointing in opposite directions. To the right of the logo is the word "KodeKloud". Below the logo, the title "SageMaker Domains and SageMaker Studio – Introduction" is displayed in white text. Underneath the title, the word "Theory" is centered. At the bottom of the video player, there is a control bar with a play button, a progress bar showing "00:00", and several small icons for volume, settings, and other media controls. The overall background of the slide is dark.

The screenshot shows a presentation slide with a dark background. On the left side, the word "Agenda" is written in white. A vertical white line runs down the center of the slide, separating it into two main sections. To the left of this line, there are four blue diamond-shaped bullet points, each containing a number and a brief description. To the right of the line, there is more white space. At the bottom left of the slide, there is a small copyright notice: "© Copyright KodeKloud". At the bottom right is the KodeKloud logo. The overall design is clean and modern.

01	Problem Statement – Limitations of standalone Jupyter notebooks
02	Solution Overview – How SageMaker Studio addresses these limitations
03	Workflow – Steps to launch SageMaker Studio
04	Results – Benefits and improvements with SageMaker Studio

Problem: Jupyter Instances Limited



© Copyright KodeKloud



Problem: Jupyter Instances Limited

01



No experiment tracking

02



Manual Git setup

03



Each user manages instances independently

04



Manual stop increases risk of extra charges

05



Limited to notebooks; needs external tools

© Copyright KodeKloud



Problem: Jupyter Instances Limited



© Copyright KodeKloud



Problem: Jupyter Instances Limited

- 01 Track experiments systematically
- 02 Record inputs and outputs
- 03 Review and select the best solution
- 04 Third-party tools add to cost (e.g., CometML)



© Copyright KodeKloud



Solution: SageMaker Domain and Studio

Domain

Elastic File System (EFS) – Shared storage (NFS v4)
User Profile – Individual workspace

Applications

SageMaker Studio – ML development environment
Canvas – No-code ML

Execution Role

IAM Role – Manages permissions

SageMaker Domain



IAM Role

User Profile

Studio

JupyterLab

R Studio

Code Editor

Studio Classic

Applications

EFS

© Copyright KodeKloud



Solution: SageMaker Domain and Studio

04

Experiment Tracking

SageMaker Experiments logs ML training runs

05

Automated Workflows

Collaborate using SageMaker Pipelines

06

Fine-Grained Access

IAM policies manage permissions for notebooks, datasets, and models

© Copyright KodeKloud



Solution: SageMaker Domain and Studio

SageMaker Domain

Acts as an administrative boundary for SageMaker Studio

Elastic File System (EFS)

Provides a shared filesystem for seamless collaboration

© Copyright KodeKloud



Workflow: SageMaker Studio

The screenshot shows the AWS SageMaker AI console. On the left, there's a sidebar with options like 'Getting started', 'Applications and IDEs' (which is expanded to show 'Studio', 'Canvas', 'RStudio', 'TensorBoard', 'Profiler', 'Notebooks', and 'Partner AI Apps'), and 'Admin configurations'. The main content area displays the 'SageMaker Studio' landing page. At the top, there's a yellow banner with a warning about JupyterLab 3 notebooks reaching end-of-support on December 31, 2024. Below the banner, the text reads: 'The first fully integrated development environment (IDE) for machine learning.' To the right, there's a 'Get Started' button with the text 'Create an Amazon SageMaker domain to use Studio and Studio Notebooks.' At the bottom of the main content area, there are sections for 'How it works', 'What is Studio?', and 'Pricing (US)'. The bottom of the screen shows the AWS navigation bar and some status icons.

Workflow: Creating a Domain

Amazon SageMaker AI > Set up SageMaker Domain

Set up SageMaker Domain

Use SageMaker Domain as the central store to manage the configuration of SageMaker for your organization.

Set up for single user (Quick setup)

Let Amazon SageMaker configure your account, and set up permissions for your SageMaker Domain.

- New IAM role with AmazonSageMakerFullAccess policy
- Public internet access, and standard encryption
- SageMaker Studio - New, and SageMaker Studio Classic integrations
- Shareable SageMaker Studio Notebooks
- SageMaker Canvas
- IAM Authentication

Perfect for single user domains and first time users looking to get started with SageMaker.

[Other domain setup options](#)

Set up for organizations

Control all aspects of account configuration, including permissions, integrations, and encryption.

- Advanced network security, and data encryption
- SageMaker Studio - New, SageMaker Studio Classic, RStudio, and Code Editor Based on Code-OSS, Visual Studio Code Open Source Integrations
- SageMaker Studio Projects, and Jumpstart
- SageMaker Canvas, and Amazon services integrations
- IAM, or IAM Identity Center (successor to AWS SSO)

Better for admins with large user groups, but you can always update your account configuration settings later if you want to do a quick setup now.

[Set up](#)

© Copyright KodeKloud



Workflow: Creating a Domain

Set up for single user (Quick setup)

Let Amazon SageMaker configure your account, and set up permissions for your SageMaker Domain.

- New IAM role with AmazonSageMakerFullAccess policy
- Public internet access, and standard encryption
- SageMaker Studio - New, and SageMaker Studio Classic integrations
- Shareable SageMaker Studio Notebooks
- SageMaker Canvas
- IAM Authentication

Perfect for single user domains and first time users looking to get started with SageMaker.

Best for testing, initial exploration, and learning SageMaker Studio

Uses the region's default VPC, which may not meet security standards

Suitable for personal development, quick proof of concepts, and non-production workloads

© Copyright KodeKloud



Workflow: Creating a Domain

Manual Setup Considerations

Security & Access Control	Infrastructure & Resources	Integration & Compliance
<ul style="list-style-type: none">Custom IAM roles and permissionsSpecific VPC configurations and security groupsCustom encryption settings (KMS keys)Specific network access controls (VPC-only vs public internet)Custom authentication methods (SAML)Specific IAM policiesCustom security controls	<ul style="list-style-type: none">Custom EBS volume sizesSpecific subnet configurationsFine-grained security group rulesCustom domain execution rolesSpecific app settings and permissionsData encryption requirementsNetwork isolation needsManual setup allows using existing VPC infrastructureBetter control over resource allocation	<ul style="list-style-type: none">Custom VPC endpointsSpecific network routingIntegration with existing enterprise systemsCustom repository configurationsAudit requirements

© Copyright KodeKloud



Workflow: Creating a Domain

When to Choose Manual Setup

- | | | | |
|----|--|----|--|
| 01 | Precise control over security settings | 04 | Cost optimization for production deployments |
| 02 | Integration with existing infrastructure | 05 | Custom network configurations |
| 03 | Compliance with organizational policies | 06 | Specific authentication methods |

© Copyright KodeKloud



Workflow: SageMaker Domain

The screenshot shows the AWS SageMaker AI console. On the left, a sidebar menu includes 'Getting started', 'Applications and IDEs' (Studio, Canvas, RStudio, TensorBoard, Profiler, Notebooks, Partner AI Apps), 'Admin configurations' (Domains, Role manager, Images, Lifecycle configurations), and 'JumpStart'. The main content area displays the 'SageMaker Studio' landing page, which features a large heading 'The first fully integrated development environment (IDE) for machine learning.', a 'Get Started' button, and sections for 'How it works' and 'Pricing (US)'. A yellow banner at the top right indicates that 'JupyterLab 3 notebooks on SageMaker Studio Classic will reach end of support on December 31, 2024. (13 days, 12 hours, 59 minutes)'.

© Copyright KodeKloud



Workflow: SageMaker Domain

The screenshot shows the AWS SageMaker AI console. The sidebar menu is identical to the previous screenshot. The main content area displays the 'Domains' management page, which includes a table titled 'Domains (1) info' with one entry: 'QuickSetupDomain-20241216T134617'. The table columns are Name, Id, Status, Created on, and Modified on. A modal window titled 'Introducing domain-level resource visibility' provides information about domains and their resources. A yellow banner at the top right indicates that 'JupyterLab 3 notebooks on SageMaker Studio Classic will reach end of support on December 31, 2024. (13 days, 12 hours, 28 minutes)'.

© Copyright KodeKloud



Workflow: SageMaker Domain

The screenshot shows the 'Domain details' page for the domain 'QuickSetupDomain-20241216T134617'. The 'General settings' section displays the domain name, creation date, status (Ready), and domain ID. The 'Domain rules' section shows a table with columns: Rule type, Application type, Rule action, and Resource.

General settings

Name	Status	Domain ID
QuickSetupDomain-20241216T134617	Ready	d-vxio2thzurak

Domain rules

Rule type	Application type	Rule action	Resource

Workflow: User Profiles

The screenshot shows the 'User profiles' page for the domain 'QuickSetupDomain-20241216T134617'. It displays a single user profile named 'default-20241216T134617' with the creation date and modified date both being 'Dec 16, 2024 22:49 UTC'. A sidebar on the right lists 'Personal apps' including Studio, Canvas, TensorBoard, and Profiler, along with a 'Collaborative Spaces' section and a 'Launch' button.

User profiles

Name	Modified on	Created on
default-20241216T134617	Dec 16, 2024 22:49 UTC	Dec 16, 2024 22:49 UTC

Workflow: User Profiles



studio-d-vxio2thzurak.studio.eu-central-1.sagemaker.aws



URL differs from the AWS Management Console



Hosted at
studio.<region>.sagemaker.aws

