Audio file

Your Recording 38.wav

Transcript

00:00:04 Speaker 2

Let's say we now want to add another user. That means adding another user profile to our domain, thereby permitting them to use Sagemaker Studio. We can do that from the User Profiles tab of our Sagemaker domain. Here I've just got my one default user for my quick setup, and if I go across to the User Profiles tab and click Add User, I'll be able to add the details of that new user. So let's do that. So I'm presented with the dialog box asking me to input the settings for that user.

00:00:29 Speaker 2

Here I'm adding a user called user 2.

00:00:32 Speaker 2

I'm then specifying what permissions I want that user to have, what Sagemaker execution role they will get that is an IAM role that determines what they are able to interact with.

00:00:42 Speaker 2

So then I'm prompted to create the settings for that user. I would give the user a name and it would specify an IAM role to be used under that user. So when that user performs activities, they will be performed under that security context.

00:00:54 Speaker 2

When I hit next, I would then be prompted to configure the applications. Now, this is a big change. Since we are deprecating Sagemaker Classic, the default selection here will be Sagemaker Studio New. Sometimes you'll hear people refer to that as Sagemaker Studio V2, but that's definitely the interface that you want, and it's the one we've been showing so far. If you had a legacy user who wanted to use the Sagemaker Studio Classic interface, that's where we would make the change for them right here. But I would encourage you to do that unless there's a very specific use case for it.

00:01:23 Speaker 2

So would you say to make a studio new now just as we are adding another user profile to our domain, we need to be aware that user profile should align to an individual person. It's a strong recommendation that a user profile directly represents a person, a user, and the security model has been designed in that way. So please don't share multiple AWS account users. Sharing the same user profile in a Sagemaker domain could lead to unintended consequences for security access. Each user profile will get a private home directory at your EFS volume if you're using Sagemaker.

00:01:53 Speaker 2

If you are using Sagemaker new, then you will be getting your own directory in an EBS volume if it is shared or you have the entire EBS volume if it is private. This user profile will maintain all your

personal settings and configurations that you make. You are using the applications made available to you in the Sagemaker domain. This way you're going to get accurate resource tracking and billing. You know which person left their Jupyter Lab space running or which person invokes the creation of an EMR cluster of 22 nodes running C524X larges and now we.

00:02:23 Speaker 2

Huge bill in our hands and ensures that any controls that are placed upon us are made user specific so that when we apply constraints, we know that they are applying to the intended recipients of those constraints. Let's summarize the best practices for creating profiles in our Sagemaker domain from the perspective of the AWS Identity Center. One Identity Center user should equal 1 Sagemaker profile. Remember, in enterprise environments we use the Identity Center for federating login from third parties like Microsoft, Entra or other SAML sources.

00:02:52 Speaker 2

We can have profiles auto created in our domain when assigned from the Identity Center to Sagemaker Studio. This will ensure that we get consistent settings and permissions for each individual user The.

00:03:02 Speaker 2

Benefits that we get from a security perspective is that we have true audit trails for understanding the actions taken by every user, we get proper isolation of the resources used by each user, and we get role-based access that makes sense for the alignment with that user profile to ensure that each user can only perform the actions that they are intended to be allowed to do.

00:03:22 Speaker 2

And let's make sure we have secure workspaces with individualized controls. From a resource management perspective, this will ensure individual resource quotas can be applied. So if we come up with a quota limiting you, let's say, to a number of EBS volumes or limiting you to a number of instances, then that will align to you rather than to a group of people. It ensures that storage space allocation is to an individual. Again, we can track how much space is being used by each individual person. If needed, we can separate the execution role, in other words, the security context.

00:03:51 Speaker 2

So if we are both users, all the same Sagemaker domain, when my Jupiter Lab space runs, maybe I have different permissions and different buckets that I can access compared to you. And we get better tracking of compute usage because we know which user created which resources and how much they're being used. So we've got far better tracking and better financial control. So we should avoid sharing user profiles between multiple people. That's not how the product has been designed. So avoid creating generic user profiles from the team. Each member of the team should have their own separate user.

00:04:22 Speaker 2

In the Sagemaker domain and we should avoid using a single profile from multiple IAM users or identity center users. Always think if it is an individual person then they should have individual user

profile in the Sagemaker domain. Now returning to the dialogue wizard that we are presented with when creating a new user, we will be presented if we chose Sagemaker Studio new as our user interface versus classic to customize our Sagemaker Studio UI. Now this is a wonderful way that we can decide does this user need to see Jupyter Lab or code editor or Canvas.

00:04:51 Speaker 2

Should I even show them Studio Classic? Probably not. If I don't have an R Studio license, don't show our studio, it's just going to confuse the user. If we're not using ML Flow or Comment ML, don't show them. So as more third party applications integrate with Sagemaker, this list will just get longer. But we've got those toggle controls per user to decide what's even exposed to them in their applications panel. If we return now to the dialogue wizard that we were presented with when adding a new user, if we chose the Sagemaker Studio new user interface option rather than Sagemaker Classic, then we've.

00:05:21 Speaker 2

Far more controls available to us to configure that user profile. What I can see here is a list of toggle switches that correspond to the available applications. So in this case here I'm choosing to show Jupyter Lab and Code Editor Canvas, but not Studio Classic. And that's a good move because Studio Classic is being deprecated. But I might decide that, well, maybe my user is not going to be using Rstudio. Maybe we don't even have a license for it. So let's maybe hide that. So don't present applications that you know you're not using. It just clutters the user interface. The whole point here is to present just what the user actually needs.

00:05:51 Speaker 2

Many third party machine learning SAS providers are now integrating the Sagemaker. You can see here we've got the Liquora Guard, Fiddler, Deep Checks, Comma, ML Flow. So again, if you're not using them, hide them and only expose them if you are using them and have a subscription to those services. And only then would they become visible to that user when they launch Sagemaker Studio. Now on the next page, we get to further customize the Sagemaker Studio user interface. The left hand navigation bar is customizable. We saw in our first look of this interface that we could see tools like the Data Wrangler or the features store or.

00:06:21 Speaker 2

Plastic map reduce clusters. Well, maybe those features are not going to be appropriate for this particular user profile. So if we don't want that option to be visible to them, take it away. Don't show them the data Wrangler or the option to interface with EMR clusters. Same under jobs. Maybe I don't want them to see training jobs, or Automl or experiments. You decide what they should gain visibility of. Now, do remember that this is purely just a visibility setting. They could still access those features via code. If you wanted to limit them from actually using them altogether. That would still need to be something defined in the permissions of the IAM role.

00:06:51 Speaker 2

The next page of the User Profile setup wizard asks about data and storage. Now this data and storage pane is a little confusing. It's referring to auto mounting EFS. But we know that EFS is a

Sagemaker Classic feature, so it's confusing to be asked about this. When we configure the user profile to use the new Sagemaker Studio new interface, which is EBS only, but it still asks you about this page. It really should be hidden at.

00:07:15 Speaker 2

We need to be aware that EFS is not required for Jupyter Lab when we're using the new studio.

00:07:21 Speaker 2

The EFS auto mount would only be needed during our user profile setup. That was for Sagemaker Classic. It's only Sagemaker Classic that needs auto mount. So now that my user profile is configured, let's go look at it in the user Profiles tab. So here we are. In the user profiles tab, I can see my new user called user 2 has been created. And if I go along to the launch button, I can then launch Studio as that user. So now that we've created our user, let's check. We have our user profile listed under user profiles and there it is. There's our user profile called user 2. And if we come to the launch button, we would then be able to.

00:07:50 Speaker 2

Launched Sagemaker Studio as that user. Now once we're into Sagemaker Studio, when we start creating Jupyter Lab spaces, we will be working in a Jupyter Lab interface. Now it might not be clear at that point whether you're working in Sagemaker Classic or whether you're working in Sagemaker Studio new. Now remember that we do storage differently in the different Jupyter Lab spaces. If it's Sagemaker Classic, then we're using EFS. If we're using Sagemaker Studio new, then we're using EDS volumes. So is a handy trick to know that if you go to the terminal of your.

00:08:21 Speaker 2

Lab space and use the terminal command DF minus H You're going to be able to see the output, your file systems and how much storage space is used. Now, if you see in the output that the file system is something like EFF something mounted on home Sagemaker user, then we know ah, right, that's using Sagemaker classic on EFS. However, if we see in the output something like slash dev nvme 1 and we know ah, that's high speed local storage to the instance, which is an EBS volume, which means we're using Sagemaker Studio new on our Jupyter space.

00:08:49 Speaker 2

So it's just something to be aware of in terms of clarifying which environment am I in. And to launch our Jupyter Lab space as a new user is just a matter of coming into the Jupyter Lab application. And now I can see, ah, I've got Jupyter Lab space 2, it's shared ah, but where's Jupyterlab space one? It's not listed because it was private to the default user. So it's not present here. I only see the shared space so far as user 2. Now I can open up that space by using the open button under action, and that will take me into my Jupyterlab and.

00:09:16 Speaker 2

To start using Jupyter notebooks as usual. What if when I created my user profile, I created it using Sagemaker Classic? How would that differ? Well in the Add user profile dialogue I have that option to pick Studio Classic, but when I do that there are fewer options available to me. But what would

happen if I created my user profile and I chose to create it as a Sagemaker Classic profile? How would things differ? Well when I was creating the new user profile and I got to configure applications, I would choose Sagemaker Studio Classic. Now this is going to limit me. I can see straight away that custom.

00:09:46 Speaker 2

Of the Studio user interface is not available to me. Remember all those toggle switches where I could decide what was visible and not visible? Can't do that in Sagemaker Classic, so it takes that option away from me. Then it asks me about notebook sharing. If I want to be able to share Jupyter notebooks, then I need to explicitly turn that on, and I need to specify an S3 location for where that occurs. And that's something we didn't have to do when we were using Sagemaker Studio New. So we've seen the option of creating a user profile as both Sagemaker Studio New and Sagemaker Classic.

00:10:16 Speaker 2

Unless there is a specific reason for Classic, always create your profiles as type Sagemaker Studio New. Sagemaker Studio New gives me many advantages and we're going to look over the next 4 slides at those advantages from 4 different perspectives. Firstly, let's consider it from a streamlined development and collaboration perspective.

00:10:32 Speaker 2

The Jupiter-based IDE gives us a unified environment. It is industry standard for data scientists for doing exploratory data analysis and for doing a machine learning development. We have shared spaces, meaning that we can have real-time collaboration between data scientists working on the same Jupyter Lab notebooks.

00:10:48 Speaker 2

We have notebook sharing now. Notebook sharing can be done either via shared links or via version control system, which most typically would recommend Git version control. And from an experiment tracking point of view, we've got the ability now to use Sagemaker experiments so that we can log what we're doing as we do it. Remember that a data scientist should be behaving like a scientist with an hypothesis, with an action, and then look at the results and compare them to earlier results. So we need some kind of experiment management framework, and Sagemaker experiments is one way we could achieve that.

00:11:18 Speaker 2

From a resource and compute management perspective, we are better using the newer Sagemaker Studio. We have on-demand kernel selection. Remember that we can have multiple tabs in Jupyterlab. Each tab could be a different Jupyter notebook and each tab could be running a different kernel. From a scaling perspective, we can have resource scaling so we can add more resources if we need them or take them away if we don't. We can also configure auto shutdown so that when we are finished with something, we know that it will safely shut itself down once we're done. That was one of the big cost flags in the old legacy notebook instances.

00:11:48 Speaker 2

And we have EBS storage now in the newer Jupyter Lab version 2. Now EBS storage will provide us with lower latency, greater throughput for our processing jobs than EFS could deliver. EFS is still there in our Sagemaker domain, but we're not using it by default. Only Sagemaker Classic uses it by default. From the perspective of machine learning operations and automation, by using Sagemaker Studio, we gain access to Sagemaker pipelines. Sagemaker pipelines give us an orchestration tool to perform a sequence of actions, but called as a single entity, like run the training pipeline.

00:12:18 Speaker 2

You can perform all the steps required or run the inference pipeline and it will perform all the steps required. We have a whole module on Sagemaker pipelines later in the course, but it is Sagemaker Studio that unlocks that functionality. We have integrated git support so that we can seamlessly auto load git repos into our notebooks. We want to strongly encourage the use of git version control for looking after our code assets.

00:12:39 Speaker 2

From a debugging and monitoring perspective, we gain access to the Sagemaker Debugger and to the Sagemaker Model Monitor. Both of these tools will help us understand our model and help us understand anything going wrong with our model when we host it. And we have easier access to host our models in Sagemaker by using Sagemaker endpoints. Remember, Sagemaker endpoints simply a way of deploying virtual machines with our models deployed inside of them. And lastly, from a security and governance perspective, Sagemaker Studio New allows us to have greater I am role-based access control with far finer.

00:13:09 Speaker 2

Control. We can control each of the different aspects of the Sagemaker Studio and what we can have the rights to use.

00:13:15 Speaker 2

We have the opportunity for better network customization. In other words, what network do the Sagemaker managed instances attached to? And if you want them only to attach to private virtual private clouds, private Vpcs that are in your accounts that limit network communication, then you can absolutely do that. And it improves auditability and logging because we've got better logging with Cloudwatch, logging all output, logging what's happening in our processing jobs, logging what's happening in our training jobs, logging what's happening in any hosted inference. Once it's in Cloudwatch logs, it then becomes searchable and potentially.

00:13:45 Speaker 2

Actionable if we set up alerts to turn that into alarms that we can be notified about or we could link remediation actions towards.

00:13:54 Speaker 2

We've seen that you need Sagemaker Studio to unlock the full functionality of the newer Sagemaker. Features like the Sagemaker Model Monitor, the Sagemaker Feature Store, the Sagemaker Model Registry, the Sagemaker Debugger, the Sagemaker Canvas, all of those features accessible only via Studio. We can't launch Sagemaker Studio user interface until we have a Sagemaker domain first. Now you are permitted multiple domains. Domains are an administrative boundary, and within a domain you define the users of that domain. Then you define for each user which applications they're allowed to use and.

00:14:24 Speaker 2

We do have the option of doing a quick start domain. A quick start domain is helpful when learning or doing proof of concept, but it is not recommended for any production environment because it uses the VPC that is default in your region and it is just really there to get you started quickly. In an enterprise environment, you're going to want to align your IAM or Identity Center users with user profiles in a domain and a manual setup would be needed for that. We've seen now that Sagemaker Studio allows us to have more than just Jupyterlab. Jupyter Lab is now one of many potential applications.

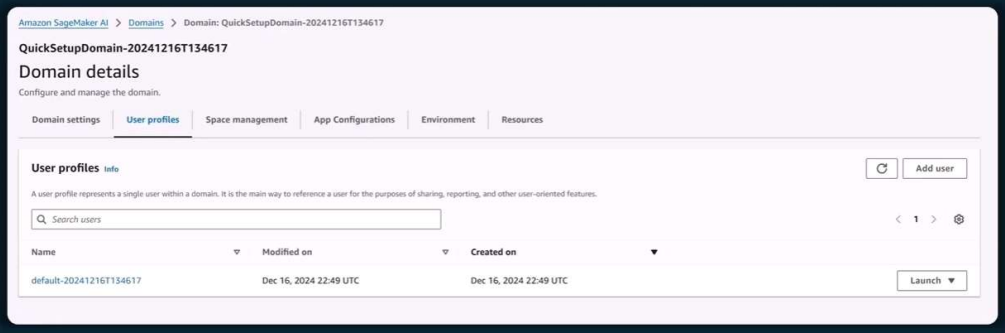
00:14:54 Speaker 2

That you can run in Sagemaker Studio environment. We also have code editor to essentially Visual Studio Code, Rstudio, ML Flow and many more third parties are now coming online exposing their applications in the interface. We've seen that to use Jupyterlab, we need to define a Jupyterlab space. That means defining the amount of compute we require, like MLT 3 Medium instance to run our Jupyter Lab server process. We've seen that when we define that space, the space can be defined as either private or shared. If it is private, it is just mine as a single user profile.

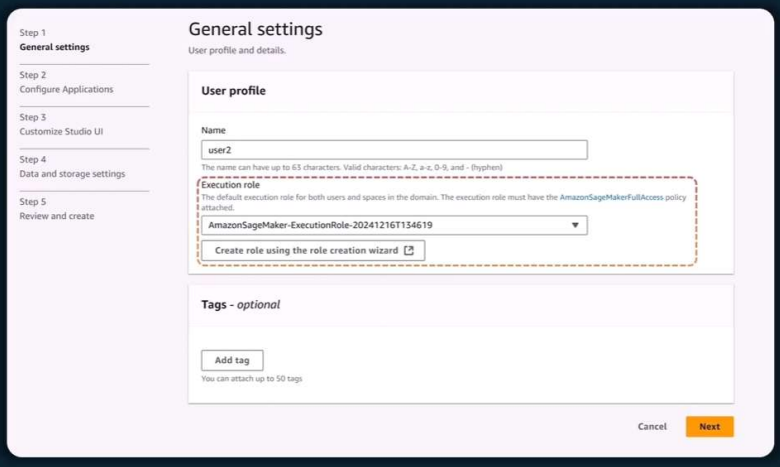
00:15:23 Speaker 2

If it is shared, it is visible and listed to all user profiles of that Sagemaker domain. We've seen Sagemaker Classic is an option when configuring a user profile, but it is very much outdated. It's no longer supported by AWS and should only be used where a customer is already using Sagemaker Classic and just needs continuity of features until they migrate to newer, better ones in Sagemaker Studio new. So that wraps up this lesson. In our next lesson, we're going to look in more detail at the options that we have for collaborating with others. See you there.

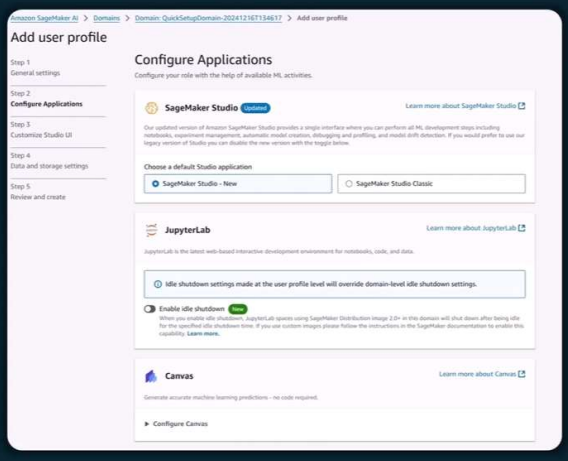
Workflow: Adding Another User



Workflow: Adding Another User



Workflow: Adding Another User



Workflow: Adding Another User

Each user profile should represent an individual person for security and efficiency.

01	02	03	04	05
Represents an **individual user** in a domain	Provides a **private home directory** (if using SageMaker Classic)	Maintains **personal settings and configurations**	Ensures **accurate resource tracking and billing**	Supports **user-specific security controls**

Workflow: Adding Another User

Security Benefits

- **Audit trails** for tracking user actions
- **Proper isolation** of user resources
- **Role-based access** for security and permissions
- **Secure workspaces** with individualized controls

Workflow: Adding Another User

Resource Management

- **Individual resource quotas** can be applied
- Personal storage space allocation
- **Separate execution roles** can be assigned if needed
- **Better tracking** of compute usage per user

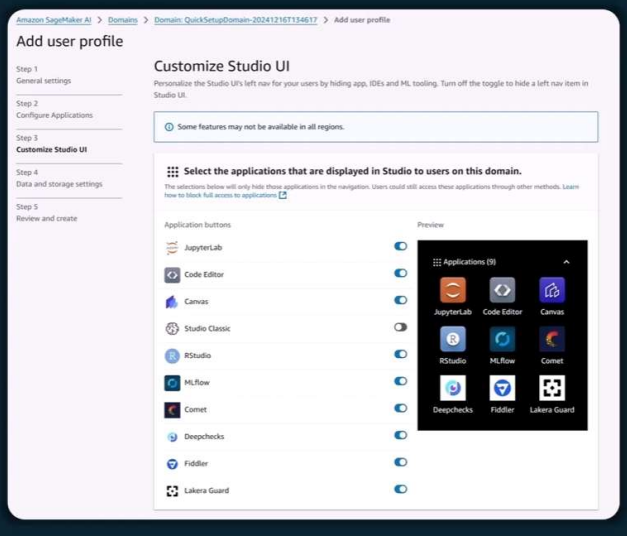
Workflow: Adding Another User

Avoid sharing user profiles between multiple people

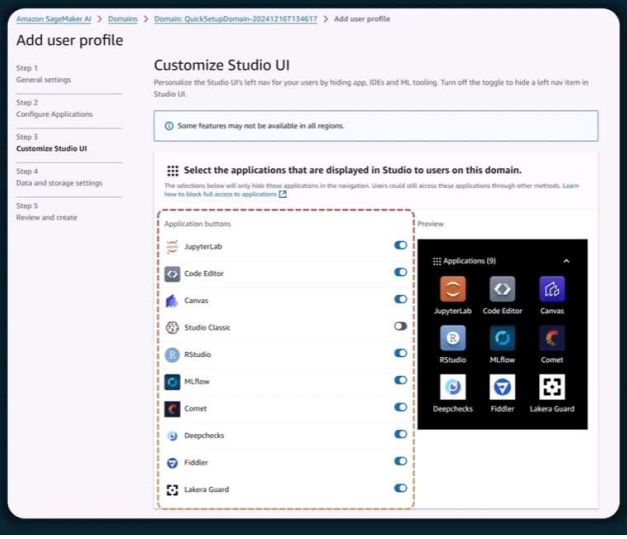
Avoid creating generic user profiles for teams

Avoid using a single user profile for multiple IAM/Identity Center users

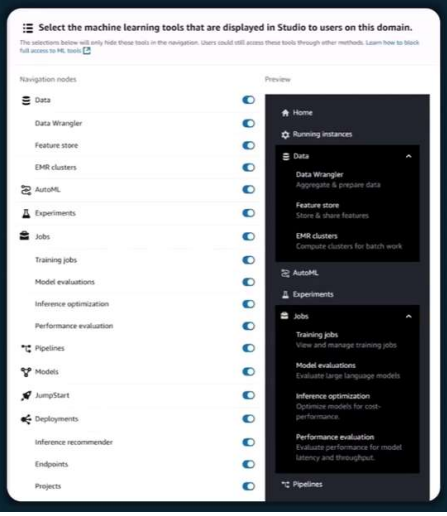
Workflow: Customizing UI



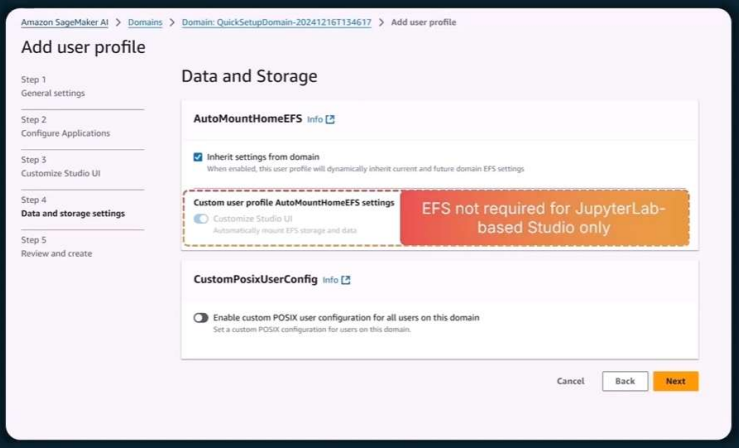
Workflow: Customizing UI



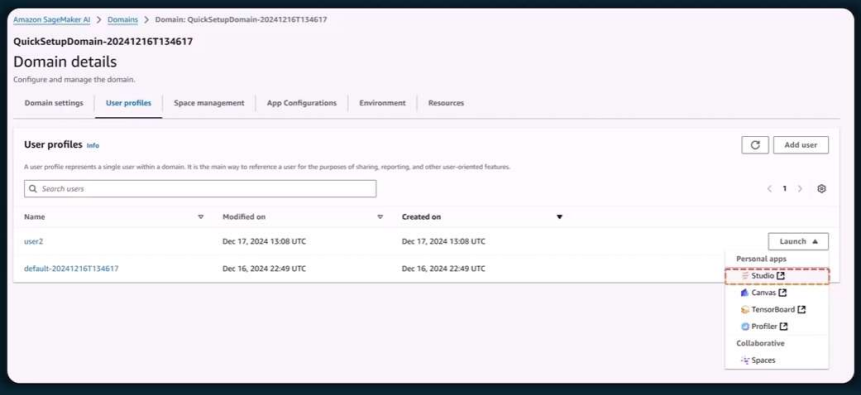
Workflow: Customizing UI



Workflow: Data and Storage



Workflow: Data and Storage

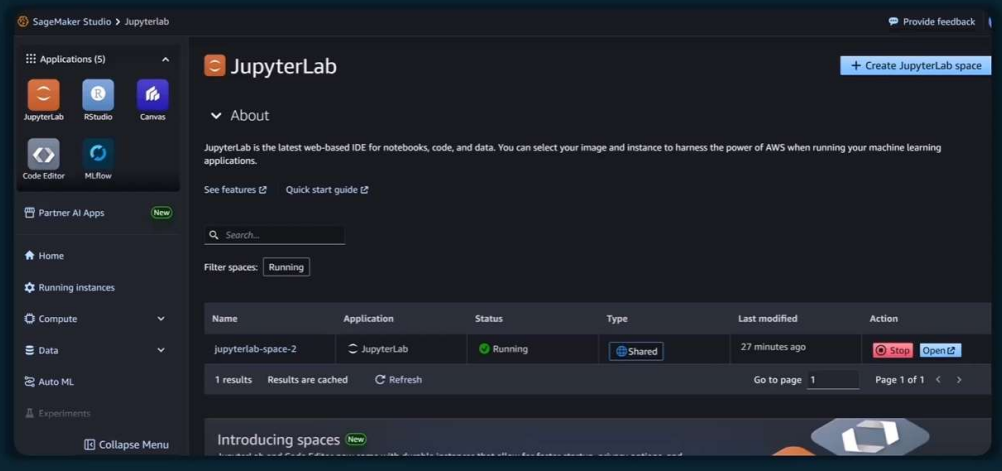


Terminal +

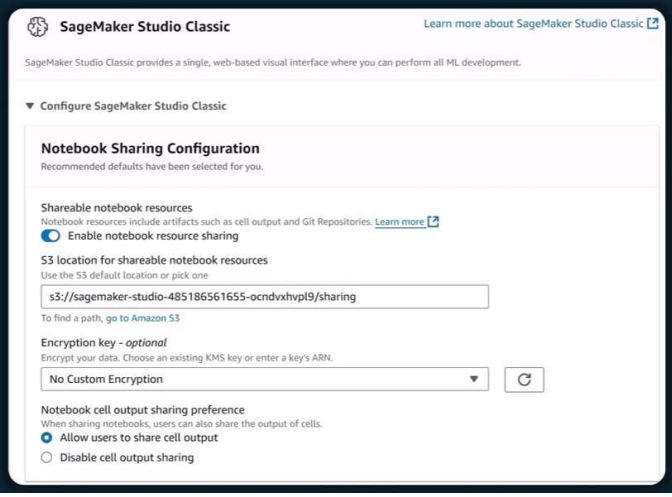
```
# Check the storage type used by SageMaker Studio
df -h
```

```
# Output
Filesystem      Size  Used Avail Use% Mounted on
/dev/nvme1n1    50G   5G   45G  10% /home/sagemaker-user
```

Workflow: Launching JupyterLab



Workflow: User Profile Classic

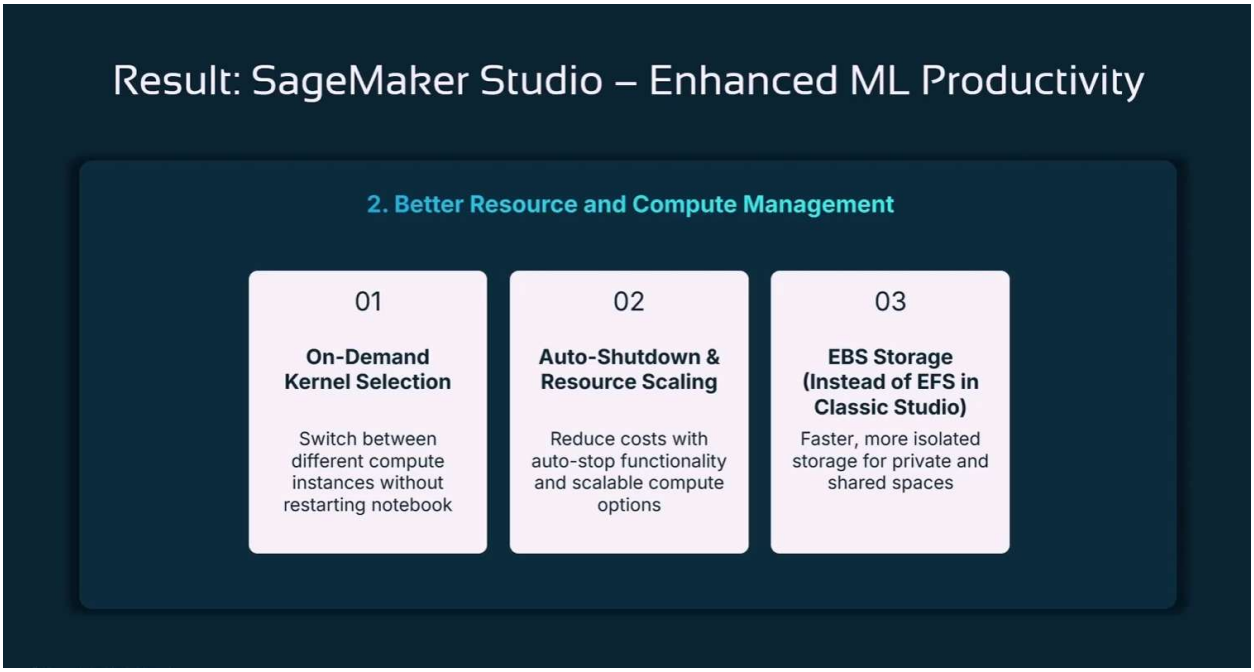


Result: SageMaker Studio – Enhanced ML Productivity

1. Streamlined Development and Collaboration

01	02	03	04
JupyterLab-Based IDE	**Shared Spaces**	**Notebook Sharing**	**Experiment Tracking**
A unified environment for coding, data exploration, and debugging	Multiple team members can collaborate in shared JupyterLab environments	Easily share notebooks via links or version control (Git)	Use SageMaker Experiments to compare models and results in an organized way

Result: SageMaker Studio – Enhanced ML Productivity



Result: SageMaker Studio – Enhanced ML Productivity

3. Improved MLOps and Automation

01	02	03	04
SageMaker Pipelines	**Integrated Git Support**	**Debugging and Monitoring**	**Easier Deployment to SageMaker Endpoints**
Automate the ML lifecycle (training, evaluation, deployment)	Seamlessly clone, push, and manage repositories inside Studio	Built-in integration with SageMaker Debugger and Model Monitor	

Result: SageMaker Studio – Enhanced ML Productivity

4. Security and Governance Improvements

01	02	03
IAM Role-Based Access Control	**Network Isolation**	**Auditability and Logging**
Manage permissions for different team members	Better VPC and security group integration for controlled access	Integrated with CloudTrail and CloudWatch for compliance

Summary

01 **SageMaker Studio** is required for newer SageMaker functionalities.

02 **Studio is launched within a domain** that defines users, applications, and storage.

03 **Quickstart domain setup** allows adding multiple users.

04 **Multiple applications available**, including JupyterLab, Code Editor, RStudio, and MLFlow.

Summary

05 **JupyterLab requires a space,** which is backed by a managed EC2 instance.

06 **Space can be private or shared**, affecting visibility.

07 **SageMaker Classic is outdated** and should not be used unless in a legacy environment.