



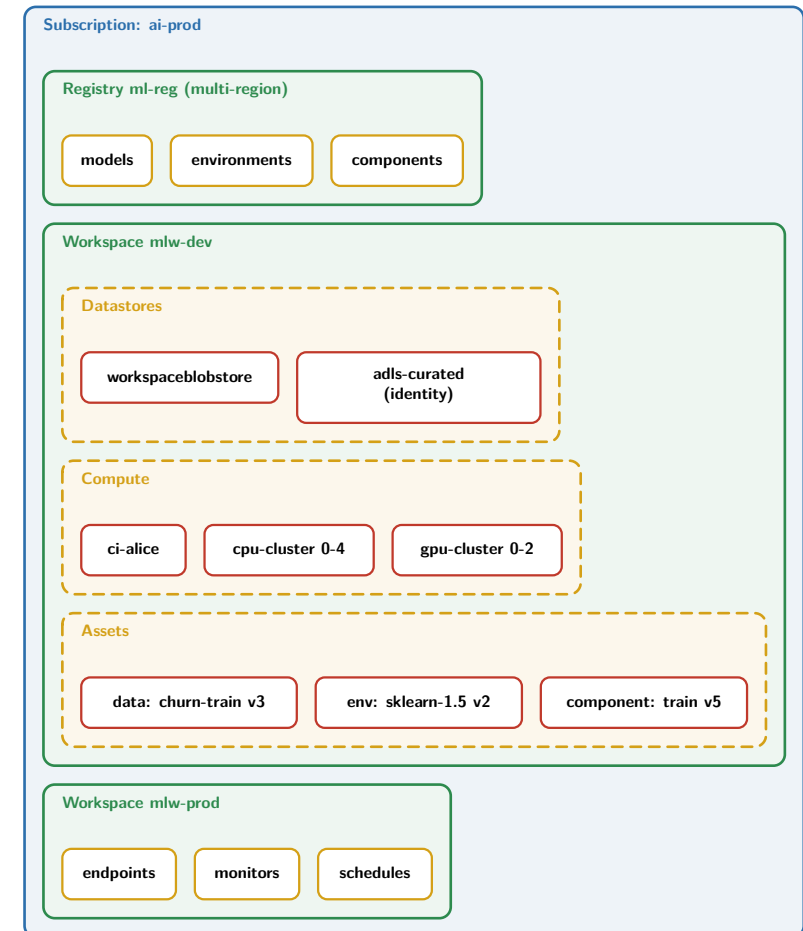
## AI-300 Domain 1: MLOps Infrastructure as Code

AI-300 skills measured, domain 1 'Design and implement an MLOps infrastructure' (15-20 %), as of 2026 | Workspace resources (datastores, compute, identity), assets (data, environments, components, registries), IaC with Bicep and Azure CLI, GitHub Actions with OIDC, network isolation, Git source control

AI-300 Domain 1 Azure ML Bicep GitHub Actions Registries

### Resources and assets in a workspace

**Recall (DP-100 sheet 01):** a workspace depends on Storage, Key Vault, Application Insights and Container Registry; data scientists use compute instances and clusters, datastores and data assets (uri\_file, uri\_folder, mltable), environments and components. **AI-300 asks you to provision and govern all of it as code and share it across teams.**



| Task                              | az ml (CLI v2) / SDK                                                                                                                               |
|-----------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| Create datastore (identity-based) | az ml datastore create -f adls.yml (no credentials block → identity access)                                                                        |
| Compute cluster scaling to 0      | az ml compute create -n cpu-cluster --type AmlCompute --size Standard_DS3_v2 --min-instances 0 --max-instances 4 --idle-time-before-scale-down 120 |
| Data asset version                | az ml data create -n churn-train -v 3 --type mltable --path ./data                                                                                 |
| Environment from image + conda    | az ml environment create -f env.yml (image + conda_file, or build context)                                                                         |
| Component                         | az ml component create -f train.yml                                                                                                                |
| Registry                          | az ml registry create -f registry.yml (locations, storage and ACR per region)                                                                      |
| Share a model to the registry     | az ml model share -n churn -v 8 --registry-name ml-reg --share-with-name churn --share-with-version 8                                              |
| Use a registry asset              | azureml://registries/ml-reg/components/train/versions/5 in job YAML                                                                                |

| Identity question                            | Answer                                                                                                           |
|----------------------------------------------|------------------------------------------------------------------------------------------------------------------|
| Who can run jobs but not create compute?     | AzureML Data Scientist at workspace scope                                                                        |
| Who manages compute only?                    | AzureML Compute Operator                                                                                         |
| Read shared assets in a registry?            | AzureML Registry User                                                                                            |
| Jobs read data without keys                  | datastore without credentials + identity: user_identity or managed identity on the cluster                       |
| Workspace accesses its storage and Key Vault | workspace managed identity (system or user-assigned; set systemDatastoresAuthMode: identity for credential-less) |
| Endpoint pulls model and reads storage       | endpoint identity (system or user-assigned) with AcrPull / Storage Blob Data Reader                              |

## ⌘ IaC with Bicep and Azure CLI

main.bicep (excerpt): private workspace with managed network

```
param env string
resource mlw 'Microsoft.MachineLearningServices/workspaces@2024-10-01' = {
  name: 'mlw-${env}'
  location: resourceGroup().location
  identity: { type: 'SystemAssigned' }
  properties: {
    storageAccount: st.id
    keyVault: kv.id
    applicationInsights: appi.id
    containerRegistry: acr.id
```

```
publicNetworkAccess: 'Disabled'
managedNetwork: { isolationMode: 'AllowOnlyApprovedOutbound' }
systemDatastoresAuthMode: 'identity'
}
}
// az deployment group what-if -g rg-ml-dev -f main.bicep -p env=dev
// az deployment group create -g rg-ml-dev -f main.bicep -p env=dev
// az ml workspace provision-network -g rg-ml-dev -n mlw-dev
```

- 0 `$ az ml job create`
- 1 `$ az ml job create -f pipelines/train.yml`  
pipeline or command job YAML
- 2 `$ az ml job create -f pipelines/train.yml -g rg-ml-dev -w mlw-dev`  
target workspace
- 3 `$ az ml job create -f pipelines/train.yml -g rg-ml-dev -w mlw-dev --set inputs.training_data.path=azureml:churn-train:3`  
override an input at submit time
- 4 `$ az ml job create -f pipelines/train.yml -g rg-ml-dev -w mlw-dev --set inputs.training_data.path=azureml:churn-train:3 --set experiment_name=churn-ci`  
group runs for comparison
- 5 `$ az ml job create -f pipelines/train.yml -g rg-ml-dev -w mlw-dev --set inputs.training_data.path=azureml:churn-train:3 --set experiment_name=churn-ci --stream`  
follow logs; non-zero exit fails the workflow

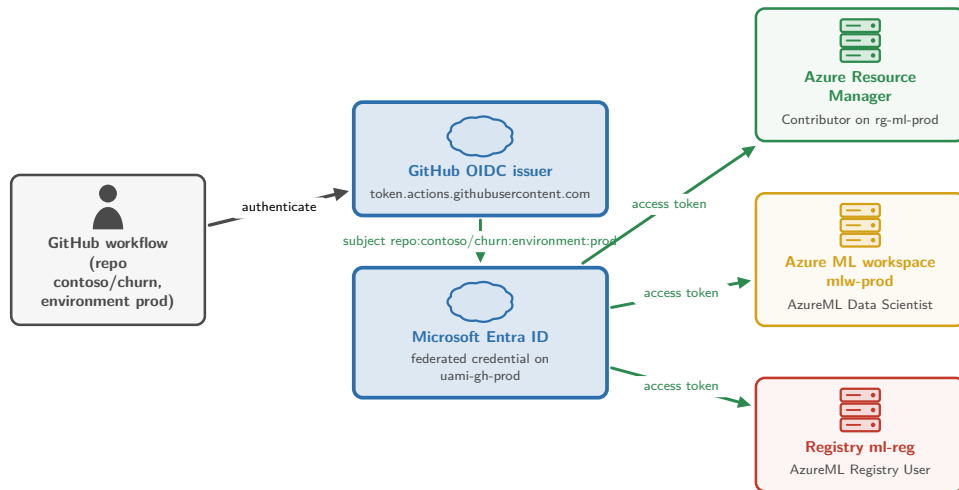
### FULL COMMAND

```
$ az ml job create -f pipelines/train.yml -g rg-ml-dev -w mlw-dev --set inputs.training_data.path=azureml:churn-train:3 --set experiment_name=churn-ci --stream
```

### 💡 TIP

Keep one Bicep module per environment layer (network, workspace, compute, registry) and parameter files per environment (dev.bicepparam, prod.bicepparam). Use what-if in PRs and create on merge.

## GitHub Actions with OIDC

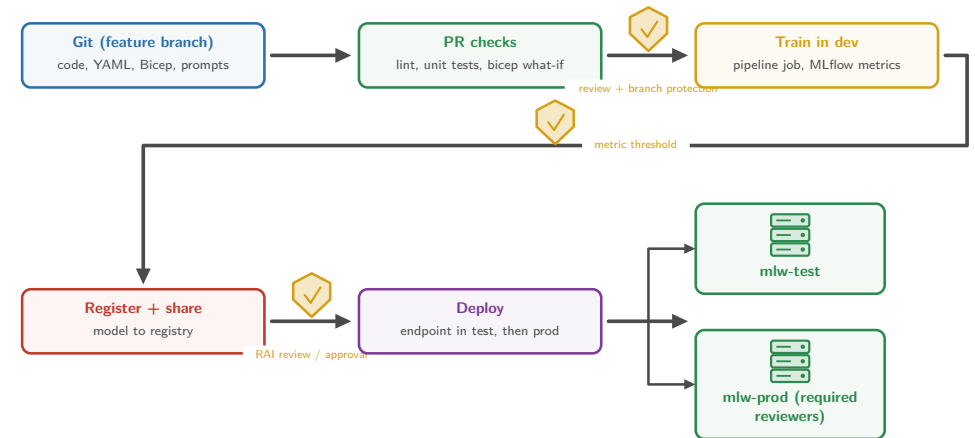


No client secret anywhere: GitHub's short-lived token is exchanged for an Entra token (audience api://AzureADTokenExchange)

```

.github/workflows/train.yml
on:
  push: {branches: [main], paths: ["src/**", "pipelines/**"]}
permissions: {id-token: write, contents: read}
jobs:
  train:
    runs-on: ubuntu-latest
    environment: prod # matches the federated subject
    steps:
      - uses: actions/checkout@v4
      - uses: azure/login@v2

    with:
      client-id: ${vars.AZURE_CLIENT_ID}
      tenant-id: ${vars.AZURE_TENANT_ID}
      subscription-id: ${vars.AZURE_SUBSCRIPTION_ID}
      - run: az extension add -n ml -y
      - run: az deployment group create -g rg-ml-prod -f infra/main.bicep -p env=prod
      - run: az ml job create -f pipelines/train.yml -g rg-ml-prod -w mlw-prod --stream
  
```



## Network isolation and Git

| Control                       | Setting                                                                                                        | Effect                                                                                    |
|-------------------------------|----------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------|
| Inbound                       | publicNetworkAccess: Disabled + private endpoint (privatelink.api.azureml.ms, privatelink.notebooks.azure.net) | studio, SDK and CLI only from the VNet / VPN                                              |
| Outbound                      | managed VNet AllowInternetOutbound                                                                             | simple; compute reaches any internet endpoint                                             |
| Outbound                      | managed VNet AllowOnlyApproved Outbound                                                                        | only private endpoint, service tag and FQDN rules (FQDN rules add an Azure Firewall cost) |
| Dependencies                  | private endpoints on Storage, Key Vault, ACR                                                                   | workspace resources not public either                                                     |
| Serverless compute Limitation | runs inside the managed VNet model monitoring does not support AllowOnlyApprovedOutbound                       | no subnet to manage choose the mode before enabling monitors (as of 2026)                 |

- **Repo layout** src/ (training code), components/, pipelines/, environments/, infra/ (Bicep), tests/, .github/workflows/; one repo per product or mono-repo with path filters
- **Branching** feature branches → PR to main (protected, reviews, checks) → tags for releases; environments test and prod with approvals
- **Traceability** jobs submitted from a Git working copy record repository, branch and commit in run properties (azureml.git.\*)
- **Never commit** secrets, data, large models (use datastores, registries, Git LFS only for small artifacts); .amignore keeps files out of job snapshots
- **Compute instance + GitHub** clone with SSH key or GitHub CLI device login on the instance

**azure/login fails: AADSTS70021 no matching federated identity record**

HIGH
COMMON

|                                                                                                                                                                    |                                                                                                                        |                                                                                                                                                                                                            |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>LIKELY CAUSES</b> <ol style="list-style-type: none"> <li>Subject mismatch (branch vs environment vs PR)</li> <li>Missing permissions id-token: write</li> </ol> | <b>HOW TO CONFIRM</b> <div> <pre>az identity federated-credential list -g rg --identity-name uami-gh-prod</pre> </div> | <b>FIX</b> <ul style="list-style-type: none"> <li>Use <code>repo:org/repo:environment:prod</code> when the job declares <code>environment: prod</code></li> <li>Add permissions id-token: write</li> </ul> |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

**Job in private workspace hangs pulling packages**

MEDIUM
OCCASIONAL

|                                                                                                                                                                                 |                                                                                                    |                                                                                                                                                        |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>LIKELY CAUSES</b> <ol style="list-style-type: none"> <li>AllowOnlyApprovedOutbound without FQDN rule for pypi.org</li> <li>Image build needs ACR private endpoint</li> </ol> | <b>HOW TO CONFIRM</b> <div> <pre>az ml workspace outbound-rule list -g rg -w mlw-prod</pre> </div> | <b>FIX</b> <ul style="list-style-type: none"> <li>Add FQDN rules (pypi.org, files.pythonhosted.org)</li> <li>Prebuild environments into ACR</li> </ul> |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|

## Worked example: cluster cost

**PROBLEM**

A CI pipeline runs 30 training jobs a day on a cluster (min 0, max 4 nodes, idle scale-down 120 s). Each job uses 1 node for 20 minutes plus about 4 minutes of node start. How many node-hours a day, and what changes with min-instances 1?

**GIVEN**

- jobs per day  $J = 30$
- run time  $r = 20$  min
- start overhead  $o = 4$  min
- idle before scale-down  $i = 2$  min
- min 1 keeps one node 24 h

1

**Billed minutes per job**

$r + o + i = 20 + 4 + 2 = 26$  min

Why: nodes bill from allocation to release

2

**Node-hours with min 0**

$30 \times 26 / 60 = 13$  node-hours

Why: scale to zero between jobs

3

**Node-hours with min 1**

24 h always on + extra nodes for overlaps

Why: saves start time, costs about 11 extra node-hours

**ANSWER**

About 13 node-hours/day with min 0; min 1 costs at least 24 node-hours/day

**CHECK**

- Use min 0 for CI clusters; use low-priority / spot VMs for retry-tolerant training

## Recap and self-test

### MYTH OR FACT?

“Store a service principal secret in GitHub”

MYTH: Use OIDC federated credentials  
No long-lived secret to leak or rotate.

### MYTH OR FACT?

“Assets can only be reused inside one workspace”

MYTH: Registries share across workspaces and regions  
Promote the exact model, environment and component versions.

### MYTH OR FACT?

“A private endpoint alone isolates the workspace”

MYTH: Outbound needs the managed VNet mode too  
And Storage, Key Vault and ACR need private endpoints.

## Key takeaways

01

Everything in YAML or Bicep

`az ml` + `az deployment group`

02

OIDC from GitHub

`subject per repo / environment, id-token: write`

03

Registries for promotion

`dev` → `registry` → `prod`

04

Identity-based data access

no keys in datastores

05

Pick the outbound mode early

approved outbound blocks monitoring

**Q1** Workflow permission needed for OIDC sign-in to Azure?

**A** `permissions: id-token: write`.

**Q2** Federated credential subject for jobs using a GitHub environment named `prod`?

**A** `repo:<org>/<repo>:environment:prod`.

**Q3** CLI to create an Azure ML registry?

**A** `az ml registry create -f registry.yml`.

**Q4** Reference a component version from a registry in YAML?

**A** `azureml://registries/<registry>/components/<name>/versions/<v>`.

**Q5** Role to run jobs without managing compute?

**A** AzureML Data Scientist.

**Q6** Two managed VNet isolation modes (besides disabled)?

**A** Allow internet outbound; allow only approved outbound.

**Q7** Preview infrastructure changes before deploying Bicep?

**A** `az deployment group what-if`.

**Q8** Cluster setting that avoids paying for idle nodes?

**A** `--min-instances 0` with a short idle scale-down time.

**Q9** File that keeps files out of an Azure ML job snapshot?

**A** `.amlignore`.