

# Operating Guide on Databricks Platform Infrastructure

Document Owner(s)	@Kevin Yang @Jay Jiang
Work Package	WP4
Status	UNDER REVIEW
Consulted	
List of Approvers	

## Table of Contents

[Scope and Source code Repository](#)

[Databricks Workspaces](#)

[Non-prod Workspaces](#)

[Prod workspaces](#)

[Steps to interact with Workspaces](#)

-  [1 Confirm or change the Ruleset in for network security-group](#)
-  [2 Interacting and provisioning with AWS Infrastructure](#)
-  [3 Provision Databricks Account-level Resources \(Cloud Resources\)](#)
-  [4 Provision Databricks Account-level Resources \(BU Resources\)](#)
  -  [Gitlab Variable provisioning](#)
-  [5 Remaining Databricks Workspace-level Resources](#)
-  [6 Populate Allowed JARs/Init Scripts to Workspace Catalogs](#)

 [Logging into Databricks workspaces](#)

 [Examining available Catalogs & associated backend for the given workspace](#)

## Scope and Source code Repository

The [IaC \(Infrastructure-as-Code\)](#) codebases consist of the five repositories to provision AWS resources, Databricks Account Resources, and Databricks workspace resources used to power Datahub-databricks platform.

In the multi-step deployment processes, the subsequent repositories may have Terraform Remote State dependencies link to its previous deployment, more details on Git Repo can be found in here

[Git Repo Topology - Data Platform Repos - Future Networks - Data Hub and Analytics Project - Confluence \(atlassian.net\)](#)

The Gitlab repository can be also summarized in the table below:

Repository name	Purposes
1. <a href="#">app-datahub-nonprod-databricks-aws-infra</a>	Underlying Databricks Infrastructure for <b>Non-prod</b> from <code>aws</code> Terraform provider, except VPC (Virtual Private Cloud Component) resources
2. <a href="#">app-datahub-prod-databricks-aws-infra</a>	Underlying Databricks Infrastructure for <b>Prod</b> from <code>aws</code> Terraform provider, except VPC (Virtual Private Cloud Component) resources following the approval based on the <b>Non-prod</b> environment
3. <a href="#">core-network-databricks-vpc-components</a>	data block for VPC and Subnets with <code>security-group</code> and <code>private-link-endpoints</code> resources for both <b>Non-prod</b> and <b>Prod</b>
4. <a href="#">databricks-unity-catalog</a>	Account-level (and workspace-agnostic) resources for <code>databricks</code> Terraform provider for both <b>prod</b> and <b>nonprod</b> ,  <i>incl. resources related for Databricks permission grants which are <a href="#">workspace agnostic despite leveraging workspace-level APIs</a>.</i>
5. <a href="#">databricks-workspaces</a>	Workspace-level for <code>databricks</code> workspace provider, namely <code>&lt;Business Unit&gt;_lab</code> or <code>&lt;Business Unit&gt;_field</code> workspaces

with the corresponding workspace ID specified via the `host` parameter.

## Databricks Workspaces

The following table summarise the workspaces provisioned within Databricks platform according to the following solution design [Infrastructure Design - Future Networks - Data Hub and Analytics Project - Confluence \(atlassian.net\)](#)

### Non-prod Workspaces

Workspace Names	Link	Purpose
<b>Non-prod:</b> digital-lab- workspace- nonprod	<a href="https://dbc-eaba2339-eb1e.cloud.databricks.com/">https://dbc-eaba2339-eb1e.cloud.databricks.com/</a>	Databricks workspaces for Digital BU in <span style="background-color: #FFD700; border: 1px solid black; padding: 2px;">LAB</span> lab environment
<b>Non-</b> <b>prod:</b> digital- field- workspace- nonprod	<a href="https://dbc-fecbb5ff-7592.cloud.databricks.com/">https://dbc-fecbb5ff-7592.cloud.databricks.com/</a>	Databricks workspaces for Digital BU in <span style="background-color: #9ACD32; border: 1px solid black; padding: 2px;">FIELD</span> environment
<b>Non-prod:</b> elec- network-lab- workspace- nonprod	<a href="https://dbc-de6c0ca1-0e35.cloud.databricks.com/">https://dbc-de6c0ca1-0e35.cloud.databricks.com/</a>	Databricks workspaces for Electricity Network BU in <span style="background-color: #FFD700; border: 1px solid black; padding: 2px;">LAB</span> environment
<b>Non-prod:</b> elec- network-field- workspace- nonprod	<a href="https://dbc-94129c9d-8f32.cloud.databricks.com/">https://dbc-94129c9d-8f32.cloud.databricks.com/</a>	Databricks workspaces for Electricity Network BU in <span style="background-color: #9ACD32; border: 1px solid black; padding: 2px;">FIELD</span> environment

### Prod workspaces

Workspace Names	Link	Purpose
Prod: digital-lab-workspace-prod	<a href="https://jemena-digital-lab.cloud.databricks.com/">https://jemena-digital-lab.cloud.databricks.com/</a>	Databricks workspaces for Digital BU in <span style="background-color: #FFD700; border: 1px solid black; padding: 2px;">LAB</span> lab environment
Prod: digital-field-workspace-prod	<a href="https://jemena-digital-field.cloud.databricks.com/">https://jemena-digital-field.cloud.databricks.com/</a>	Databricks workspaces for Digital BU in <span style="background-color: #9ACD32; border: 1px solid black; padding: 2px;">FIELD</span> environment
Prod: elec-network-lab-workspace-prod	<a href="https://jemena-elec-network-lab.cloud.databricks.com/">https://jemena-elec-network-lab.cloud.databricks.com/</a>	Databricks workspaces for Electricity Network BU in <span style="background-color: #FFD700; border: 1px solid black; padding: 2px;">LAB</span> environment
Prod: elec-network-field-workspace-prod	<a href="https://jemena-elec-network-field.cloud.databricks.com/">https://jemena-elec-network-field.cloud.databricks.com/</a>	Databricks workspaces for Electricity Network BU in <span style="background-color: #9ACD32; border: 1px solid black; padding: 2px;">FIELD</span> environment

## Steps to interact with Workspaces

### 1 Confirm or change the Ruleset in for network security-group

As first point of interactions, navigate and clone [Jemena / Projects / Future Networks Databricks / core-network-databricks-vpc-components · GitLab](https://gitlab.jemena.com/jemena/projects/future-networks-databricks/core-network-databricks-vpc-components) to your IDE, you would see the repository structure exhibits the following, where it contains both

- the `security-groups` configured for ingress/egress for traffics for Databricks connectivity between the **control plane** and the **data plane**
- the `private-link` resources used to deploy Private VPC endpoint as per security compliance for private connectivity

i Databricks operates out of a **control plane** and a **compute plane**.

- The **control plane** includes the backend services that Databricks manages in your Databricks account. The web application is in the control plane.

- The **compute plane** is where your data is processed. There are two types of compute planes depending on the compute that you are using.

```

1  └── README.md
2  └── backend.tf
3  └── main-frontend-privateLink.tf
4  └── main-npd-vpc-resources.tf
5  └── main-prod-vpc-resources.tf
6  └── modules
7      └── private-link-endpoints
8          ├── README.md
9          ├── main.tf
10         ├── outputs.tf
11         ├── variables.tf
12         └── versions.tf
13     └── security-group
14         ├── README.md
15         ├── main.tf
16         ├── outputs.tf
17         ├── variables.tf
18         └── versions.tf
19     └── output.tf
20     └── providers.tf
21     └── shared-locals.tf
22     └── shared.auto.tfvars
23     └── variables.tf

```

**⚠** Please make changes via a **feature branch** before submitting a merge request to main branch. CI/CD is configured to perform

- `terraform plan` on feature branch
- and both `terraform plan` and `terraform apply` on the main branch

*navigate to `./shared_locals.tf`, and append the ingress/egress rules to the existing `data.aws_subnet`, by declaring a variable in the `locals` block, in the syntax of `allow<port>`*

```

1 # Allowing 2443 bidirectional traffic
2 allow2443 = [for cidr in values(data.aws_subnet.nonprod_subnets).*.cidr_block :
3     {
4         rule_description = "Allow cluster bidirectional connectivity on 2443"
5         ip_protocol     = "tcp"
6         from_port       = 2443
7         to_port         = 2443
8         cidr_ipv4      = "${cidr}"
9     }
10 ]

```

then, navigate to `<nonprod/prod>_vpc_resources.tf` for `nonprod` or `prod_vpc_resources.tf` for `prod`, and append the port to your `nonprod_dataplane_sg_config` variable nested in `locals` block.

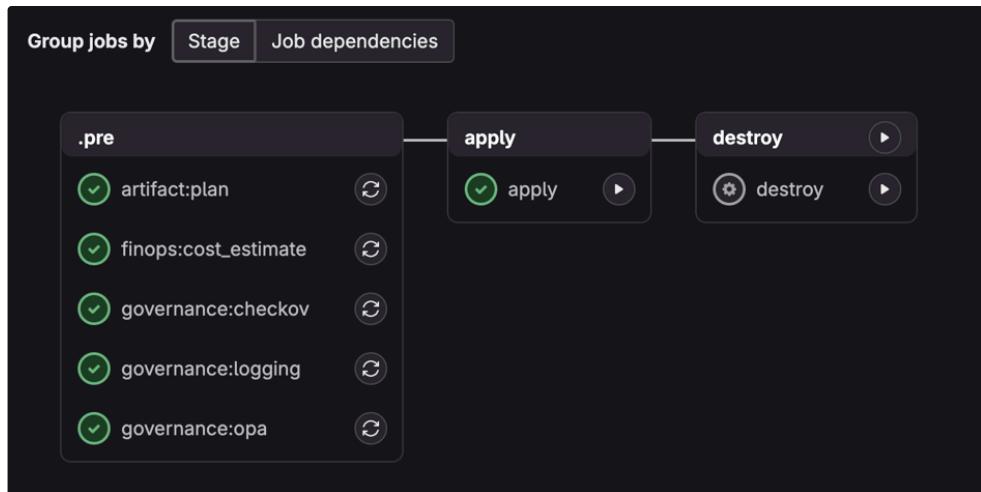
For the example below, ruleset such as `local.allow443`, `local.allow6666`, `local.allow2443` has been appended into the terraform scripts, for bidirectional traffic as per defined for [backend private-link connectivity \(Databricks Official Doc\)](#). You may customize single direction via appending only to `egress` and `ingress` rules.

```
1 nonprod_dataplane_sg_config = {
2     "dataplane-tcp-private-links" = {
3         create_sg      = true
4         name          = "${var.nonprod_prefix}-sg-dataplane-tcp-bidirectional"
5         description   = "Common bidirectional security group for databricks nonprod VPC private
links"
6         egress        = concat(local.allow443, local.allow6666, local.allow2443)
7         ingress       = concat(local.allow443, local.allow6666, local.allow2443)
8     }
9 }
```

on the same directory, comment out the `nonprod_private_link_endpoints` and/or `prod_private_link_endpoints`, as we will want to provision ENI after configuring network traffic:

```
1 # Comment out the following
2 module "nonprod_private_link_endpoints" {
3     depends_on      = [module.nonprod_databricks_common_workspace_dataplane_sg]
4     source          = "./modules/private-link-endpoints/"
5     prefix          = var.nonprod_prefix
6     vpc_id          = var.core_network_shared_nonprod_vpc_id
7     eni_subnet_names = var.nonprod_eni_subnet_names
8     security_group_ids = [
9         module.nonprod_databricks_common_workspace_dataplane_sg.security_group_ids.dataplane-
tcp-private-links
10    ]
11    tags = local.nonprod_additional_tags
12 }
```

apply the resources onto the AWS platform after GitLab gitlab merge request, where the CI/CD pipeline would've been setout to perform the following stages (with apply that can be manually triggered via  button shown below).



Upon completion, `security-group`s and for `prod/non-prod` would've been completed.

navigating back to `<nonprod/prod>_vpc_resources.tf`, and comment the following:

```

1 # Uncomment the following
2 module "nonprod_private_link_endpoints" {
3   depends_on      = [module.nonprod_databricks_common_workspace_dataplane_sg]
4   source          = "./modules/private-link-endpoints/"
5   prefix          = var.nonprod_prefix
6   vpc_id          = var.core_network_shared_nonprod_vpc_id
7   eni_subnet_names = var.nonprod_eni_subnet_names
8   security_group_ids = [
9     module.nonprod_databricks_common_workspace_dataplane_sg.security_group_ids.dataplane-
10    tcp-private-links
11  ]
12  tags = local.nonprod_additional_tags
}

```

and follow the plan/apply steps above, `private-link-endpoints` will be up.

## 2 Interacting and provisioning with AWS Infrastructure

Depending on your AWS environment (Prod/Non-prod), navigate to either

- **Non-prod:** [Jemena / Projects / Future Networks Datahub / Databricks / app-datahub-nonprod-databricks-aws-infra · GitLab](#)
- **Prod:** [Jemena / Projects / Future Networks Datahub / Databricks / app-datahub-prod-databricks-aws-infra · GitLab](#)

where the directory is structured (`nonprod` as example) as follows:

✓ View databricks-aws-infra structures

```

1 └── README.md
2 └── backend.tf

```

```
3 └── main-account-data-egress.tf
4 └── main-account-log-storage.tf
5 └── main-account-metastore-storage.tf
6 └── main-account-metastore-uc-role.tf
7 └── main-digital-bu-catalog-cloud-resources.tf
8 └── main-digital-bu-wks-cloud-resources.tf
9 └── main-elec-network-bu-catalog-cloud-resources.tf
10 └── main-elec-network-bu-wks-cloud-resources.tf
11 └── main-secrets.tf
12 └── modules
13     ├── bucket
14         ├── README.md
15         ├── default_policies.tf
16         ├── locals.tf
17         ├── main.tf
18         ├── outputs.tf
19         └── variables.tf
20     ├── catalog-cloud-resources
21         ├── catalog_buckets.tf
22         ├── ext_tbl_access_role.tf
23         ├── locals.tf
24         ├── output.tf
25         ├── uc_data_access_role.tf
26         └── variables.tf
27     ├── metastore-iam-role
28         ├── main.tf
29         ├── outputs.tf
30         └── variables.tf
31     ├── secrets
32         ├── main.tf
33         ├── outputs.tf
34         └── variables.tf
35     ├── security-group
36         ├── README.md
37         ├── main.tf
38         ├── outputs.tf
39         ├── variables.tf
40         └── versions.tf
41     └── workspace-cloud-resources
42         ├── README.md
43         ├── bucket-policies.tf
44         ├── buckets.tf
45         ├── cross-account-medallion-pass-role.tf.skip
46         ├── cross-account-role.tf
47         ├── locals.tf
48         ├── outputs.tf
49         ├── security-groups.tf
50         └── variables.tf
51 └── (nonprod/nonprod).auto.tfvars
52 └── outputs.tf
53 └── providers.tf
54 └── shared-data.tf
55 └── shared-locals.tf
56 └── tags.tf
57 └── variables.tf
```

For creating the BU resources, first navigate to or create `main-<Business Unit>-bu-catalog-cloud-resources.tf`, and define the syntax like the following, and substituting the `<>` with the related parameters:

```
1 module "digital_bu_catalog_cloud_resources" {
2   for_each = toset(["lab", "qa", "field"])
3   source   = "./modules/catalog-cloud-resources"
4
5   environment           = var.environment
6   aws_account_id        = var.aws_account_id
7   aws_region             = "ap-southeast-2"
8   databricks_account_id = var.databricks_account_id
9   stage_storage_credential_external_id = var.databricks_account_id //placeholder
10  databricks_control_plane_aws_account_id = var.databricks_control_plane_aws_account_id
11  bu_kms_key_arn         = var.digital_kms_key_arn
12  business_unit          = "digital"
13  bucket_name_prefix     = "${var.prefix}-${var.environment}"
14  data_stage              = each.value
15 }
16
17 }
```

Subsequently, navigate to or create `main-<Business Unit>-bu-wks-cloud-resources.tf`, and edit the following:

```
1 locals {
2   <Business Unit>_bu_lab_workspace_name    = "<Business Unit>-lab-
3   workspace-${var.environment}"
4   <Business Unit>_bu_field_workspace_name = "<Business Unit>-field-
5   workspace-${var.environment}"
6
7   <Business Unit>_bu_lab_workspace_buckets = {
8     "root" = "${var.prefix}-s3-${local.digital_bu_lab_workspace_name}-root"
9   }
10  <Business Unit>_bu_field_workspace_buckets = {
11    "root" = "${var.prefix}-s3-${local.digital_bu_field_workspace_name}-root"
12  }
13 }
```

and assigning the following variable at `(nonprod/prod).auto.tfvars`:

- ⓘ Note that the underlying `aws_account_id` (prod or non-prod) is a member account that sits on a larger Organizational Units (OU, i.e. `core_network_aws_account_id`). Refer to this [AWS documentation](#) for more info.

The resources coming from OU and requires platform team's implementation incl. but not limited to:

- metastore\_kms\_key
- private\_eni\_subnet\_names
- <Business Unit>\_kms\_key\_arn

```

1 environment                      = <prod/nonprod>
2 target_vpc_id                   = ""
3 databricks_account_id           = ""
4 databricks_control_plane_aws_account_id = ""
5 aws_account_id                  = ""
6 core_network_aws_account_id    = ""

8 #kms_alias: arn:aws:kms:ap-southeast-2:<account ID>:alias/app-datahub-<nonprod/prod>-
9 databricks-metastore-kms-key
9 metastore_kms_key_arn = "arn:aws:kms:ap-southeast-2:<account ID>:key/f8cee53a-4749-4935-
9 f17-fcf#####"
10 private_eni_subnet_names = [
11   "app-datahub-dev-eni-az<#>-<#1>" # e.g. "app-datahub-dev-eni-az1-01"
12   , "app-datahub-dev-eni-az<#>-<#2>"
13   , "app-datahub-dev-eni-az<#>-<#3>"
14 ]
15
16 #app-datahub-<nonprod/prod>-databricks-digital-kms-key
17 digital_kms_key_arn = "arn:aws:kms:ap-southeast-2:<account ID>:key/e0#####-b#####-####-#####
17 #####"
18 #app-datahub-<nonprod/prod>-databricks-elec-network-kms-key
19 elec_network_kms_key_arn = "arn:aws:kms:ap-southeast-2:<account ID>:key/61f#####-b#####-#####
19 #####-#####"
20
21 #app-datahub-<nonprod/prod>-databricks-<Business Unit>-kms-key
22 ...
23

```

Whilst provisioning the cloud resources, referred to the pipeline below to identify required edits to pass the plug-in, e.g. `checkov`, `opa`, and `logging`.

**⚠ DO NOT trigger `destroy` as it may have impact to the subsequent provisioning**

For `checkov` [warnings](#) related to legacy Terraform version, add the following comment format within the resource block to resolve specific warning number i.e.

`CKV2_AWS_<#>`

*For example, for disabling S3 bucket warning due to standalone public access or lifecycle configuration not been configure as S3 attribute (Optional, Deprecated):*

```

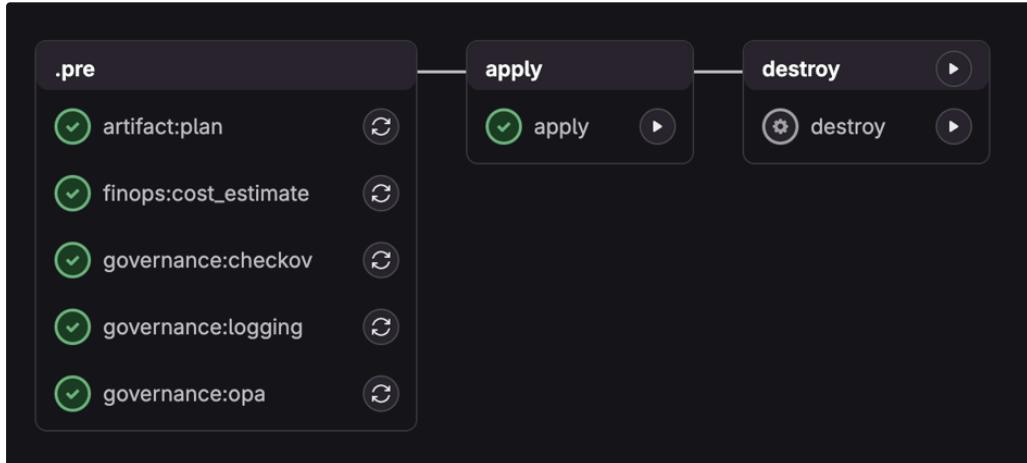
1 resource "aws_s3_bucket" "this" {
2   #checkov:skip=CKV2_AWS_61: lifecycle configuration declared in the resource block
   below

```

```

3   #checkov:skip=CKV2_AWS_6: S3 bucket Public Access Block declared in the resource
4   block below
5
6 }

```



Cloud resource CI/CD workflows

Lastly, click through the **apply** text to view the outputs, take note of the structure as you may need to pass your terraform state files, with example below:

```

1 Releasing state lock. This may take a few moments...
2 Apply complete! Resources: 0 added, 6 changed, 0 destroyed.
3 Outputs:
4 digital_bu_catalog_cloud_resources = {
5   "field_catalog" = {
6     "bronze" = {
7       "arn" = "arn:aws:s3:::app-datahub-prod-s3-digital-field-bronze"
8       "name" = "app-datahub-nonprod-s3-digital-field-bronze"
9     }
10    "data_access_iam_role" = {
11      "arn" = "arn:aws:iam:<account-ID>:role/digital-field-ctlg-uc-data-access"
12      "name" = "digital-field-ctlg-uc-data-access"
13    }
14  ...
15 }

```

### 3 Provision Databricks Account-level Resources (Cloud Resources)

Following provisioning on AWS provider, head to [Jemena / Projects / Future Networks Datahub / Databricks / databricks-unity-catalog](#) to further provisioning resources.

To kickstart Databricks platform, examine the following repository structure (dropdown):

View databricks-unity-catalog structure

```

1 |   README.md
2 |   modules

```

```
3   └── log-delivery
4       ├── main-log-delivery.tf
5       ├── outputs.tf
6       ├── providers.tf
7       └── variables.tf
8   └── managed-workspace
9       ├── README.md
10      ├── outputs.tf
11      ├── providers.tf
12      ├── variables.tf
13      └── wks_cmk.tf
14          └── wks_config.tf
15   └── name
16       ├── README.md
17       ├── example
18           └── test_names.tf
19       └── main.tf
20   └── rbac-hierarchy
21       ├── README.md
22       ├── main-access-groups.tf
23       ├── main-functional-groups.tf
24       ├── main-workspace-access.tf
25       ├── main-workspace-service-principal.tf
26       ├── outputs.tf
27       ├── providers.tf
28       └── variables.tf
29   └── service-principal-gitlab
30       ├── README.md
31       ├── data.tf
32       ├── main-gitlab-var.tf
33       ├── outputs.tf
34       ├── providers.tf
35       └── variables.tf
36   └── scripts
37       ├── metastore-destroy.sh
38       ├── tf-apply.sh
39       └── tf-plan.sh
40   └── stacks
41       ├── cloud-resources
42           ├── backend.tf
43           ├── main.tf
44           ├── outputs.tf
45           ├── providers.tf
46           ├── upstream_stacks.tf
47           └── variables.tf
48       └── digital-bu
49           ├── backend.tf
50           ├── dependencies.tf
51           ├── locals.tf
52           ├── main-bu-cred.tf
53           ├── main-catalog-field.tf
54           ├── main-catalog-lab.tf
55           ├── main-system-schema.tf
56           ├── main-workspace-field.tf
57           ├── main-workspace-lab.tf
58           ├── main-xacc-resources.tf
59           ├── outputs.tf
60           └── providers.tf
```

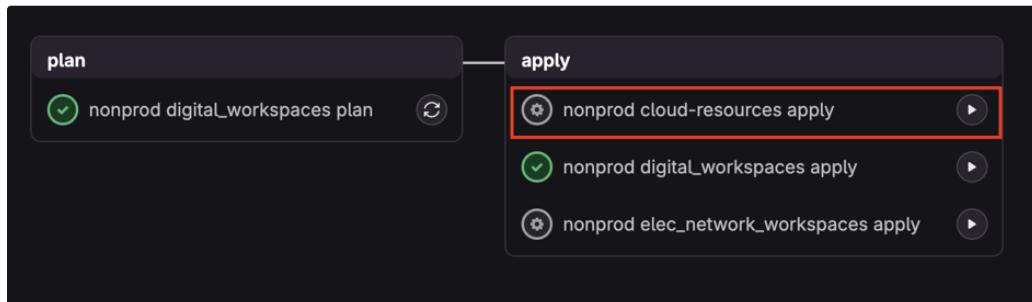
```

61      └── sim-functional-groups.tf
62      └── sim-group-membership.tf
63      └── variables.tf
64  elec-network-bu
65      ├── backend.tf
66      ├── dependencies.tf
67      ├── locals.tf
68      ├── main-bu-cred.tf
69      ├── main-catalog-field.tf
70      ├── main-catalog-lab.tf
71      ├── main-common-grants.tf
72      ├── main-system-schema.tf
73      ├── main-workspace-field.tf
74      ├── main-workspace-lab.tf
75      ├── outputs.tf
76      ├── providers.tf
77      ├── repair.tf
78      ├── sim-functional-groups.tf
79      ├── sim-group-membership.tf
80      └── variables.tf
81  metastore
82      ├── backend.tf
83      ├── main.tf
84      ├── outputs.tf
85      ├── providers.tf
86      ├── upstream_stacks.tf
87      └── variables.tf

```

**⚠ Note that `metastore apply` has been disabled to allow idempotent operations for other account-wide activity. i.e. edit `./stacks/metastore` with cautions**

With `rest_backend` and `scc_relay` provisioned via `cloud-resources`, simply re-run cloud resource applies via existing CI/CD setup:



#### 4 Provision Databricks Account-level Resources (BU Resources)

BU workspace provisioning occurs via stacks repository (browse through `digital-bu` and `elec-network-bu` for setup example).

**ℹ NOTE** that `grants.tf` and `catalog.tf` and any Terraform blocks associated with

- `databricks.workspace`
- `databricks.lab_mws`
- `databricks.field_mws`

will undergo after workspaces resources are provisioned.

Specifically, `./stacks/<BU-stack-folder>/locals.tf` has been setup for the respective business unit workspace that fetches the resources previously provisioned.

For `subnets` ID mapping to workspace and corresponding environment, refer to the following page for more information [Infrastructure Design | VPC & Subnets](#)

```

1  locals {
2    business_unit = <To be filled-in>
3
4    //account level resources
5
6    target_vpc_id = {
7      nonprod = "vpc-0ff11056a20a2ce44"
8      prod   = "vpc-002f643f2c6498c81"
9    }
10
11   scc_eni_id = {
12     nonprod =
13       data.terraform_remote_state.nonprod_mws_endpoints.outputs.scc_relay_endpoint_id
14     prod   = data.terraform_remote_state.prod_mws_endpoints.outputs.scc_relay_endpoint_id
15   }
16
17   rest_eni_id = {
18     nonprod =
19       data.terraform_remote_state.nonprod_mws_endpoints.outputs.rest_backend_endpoint_id
20     prod   =
21       data.terraform_remote_state.prod_mws_endpoints.outputs.rest_backend_endpoint_id
22   }
23
24   metastore_id = {
25     nonprod = data.terraform_remote_state.nonprod_metastore.outputs.metastore_id
26     prod   = data.terraform_remote_state.prod_metastore.outputs.metastore_id
27   }
28
29   metastore_admin_gid = {
30     nonprod = data.terraform_remote_state.nonprod_metastore.outputs.metastore_admin_group_id
31     prod   = data.terraform_remote_state.prod_metastore.outputs.metastore_admin_group_id
32   }
33
34   locals {
35     // workspace level resources
36     workspace_config = {
37       nonprod = {
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
59
60
61
62
63
64
65
66
67
68
69
69
70
71
72
73
74
75
76
77
78
79
79
80
81
82
83
84
85
86
87
88
89
89
90
91
92
93
94
95
96
97
98
99
99
100
101
102
103
104
105
106
107
108
109
109
110
111
112
113
114
115
116
117
118
119
119
120
121
122
123
124
125
126
127
128
129
129
130
131
132
133
134
135
136
137
138
139
139
140
141
142
143
144
145
146
147
148
149
149
150
151
152
153
154
155
156
157
158
159
159
160
161
162
163
164
165
166
167
168
169
169
170
171
172
173
174
175
176
177
178
179
179
180
181
182
183
184
185
186
187
187
188
189
189
190
191
192
193
194
195
196
197
197
198
199
199
200
201
202
203
204
205
206
207
207
208
209
209
210
211
212
213
214
215
215
216
217
217
218
219
219
220
221
222
223
223
224
225
225
226
227
227
228
229
229
230
231
231
232
232
233
233
234
234
235
235
236
236
237
237
238
238
239
239
240
240
241
241
242
242
243
243
244
244
245
245
246
246
247
247
248
248
249
249
250
250
251
251
252
252
253
253
254
254
255
255
256
256
257
257
258
258
259
259
260
260
261
261
262
262
263
263
264
264
265
265
266
266
267
267
268
268
269
269
270
270
271
271
272
272
273
273
274
274
275
275
276
276
277
277
278
278
279
279
280
280
281
281
282
282
283
283
284
284
285
285
286
286
287
287
288
288
289
289
290
290
291
291
292
292
293
293
294
294
295
295
296
296
297
297
298
298
299
299
300
300
301
301
302
302
303
303
304
304
305
305
306
306
307
307
308
308
309
309
310
310
311
311
312
312
313
313
314
314
315
315
316
316
317
317
318
318
319
319
320
320
321
321
322
322
323
323
324
324
325
325
326
326
327
327
328
328
329
329
330
330
331
331
332
332
333
333
334
334
335
335
336
336
337
337
338
338
339
339
340
340
341
341
342
342
343
343
344
344
345
345
346
346
347
347
348
348
349
349
350
350
351
351
352
352
353
353
354
354
355
355
356
356
357
357
358
358
359
359
360
360
361
361
362
362
363
363
364
364
365
365
366
366
367
367
368
368
369
369
370
370
371
371
372
372
373
373
374
374
375
375
376
376
377
377
378
378
379
379
380
380
381
381
382
382
383
383
384
384
385
385
386
386
387
387
388
388
389
389
390
390
391
391
392
392
393
393
394
394
395
395
396
396
397
397
398
398
399
399
400
400
401
401
402
402
403
403
404
404
405
405
406
406
407
407
408
408
409
409
410
410
411
411
412
412
413
413
414
414
415
415
416
416
417
417
418
418
419
419
420
420
421
421
422
422
423
423
424
424
425
425
426
426
427
427
428
428
429
429
430
430
431
431
432
432
433
433
434
434
435
435
436
436
437
437
438
438
439
439
440
440
441
441
442
442
443
443
444
444
445
445
446
446
447
447
448
448
449
449
450
450
451
451
452
452
453
453
454
454
455
455
456
456
457
457
458
458
459
459
460
460
461
461
462
462
463
463
464
464
465
465
466
466
467
467
468
468
469
469
470
470
471
471
472
472
473
473
474
474
475
475
476
476
477
477
478
478
479
479
480
480
481
481
482
482
483
483
484
484
485
485
486
486
487
487
488
488
489
489
490
490
491
491
492
492
493
493
494
494
495
495
496
496
497
497
498
498
499
499
500
500
501
501
502
502
503
503
504
504
505
505
506
506
507
507
508
508
509
509
510
510
511
511
512
512
513
513
514
514
515
515
516
516
517
517
518
518
519
519
520
520
521
521
522
522
523
523
524
524
525
525
526
526
527
527
528
528
529
529
530
530
531
531
532
532
533
533
534
534
535
535
536
536
537
537
538
538
539
539
540
540
541
541
542
542
543
543
544
544
545
545
546
546
547
547
548
548
549
549
550
550
551
551
552
552
553
553
554
554
555
555
556
556
557
557
558
558
559
559
560
560
561
561
562
562
563
563
564
564
565
565
566
566
567
567
568
568
569
569
570
570
571
571
572
572
573
573
574
574
575
575
576
576
577
577
578
578
579
579
580
580
581
581
582
582
583
583
584
584
585
585
586
586
587
587
588
588
589
589
590
590
591
591
592
592
593
593
594
594
595
595
596
596
597
597
598
598
599
599
600
600
601
601
602
602
603
603
604
604
605
605
606
606
607
607
608
608
609
609
610
610
611
611
612
612
613
613
614
614
615
615
616
616
617
617
618
618
619
619
620
620
621
621
622
622
623
623
624
624
625
625
626
626
627
627
628
628
629
629
630
630
631
631
632
632
633
633
634
634
635
635
636
636
637
637
638
638
639
639
640
640
641
641
642
642
643
643
644
644
645
645
646
646
647
647
648
648
649
649
650
650
651
651
652
652
653
653
654
654
655
655
656
656
657
657
658
658
659
659
660
660
661
661
662
662
663
663
664
664
665
665
666
666
667
667
668
668
669
669
670
670
671
671
672
672
673
673
674
674
675
675
676
676
677
677
678
678
679
679
680
680
681
681
682
682
683
683
684
684
685
685
686
686
687
687
688
688
689
689
690
690
691
691
692
692
693
693
694
694
695
695
696
696
697
697
698
698
699
699
700
700
701
701
702
702
703
703
704
704
705
705
706
706
707
707
708
708
709
709
710
710
711
711
712
712
713
713
714
714
715
715
716
716
717
717
718
718
719
719
720
720
721
721
722
722
723
723
724
724
725
725
726
726
727
727
728
728
729
729
730
730
731
731
732
732
733
733
734
734
735
735
736
736
737
737
738
738
739
739
740
740
741
741
742
742
743
743
744
744
745
745
746
746
747
747
748
748
749
749
750
750
751
751
752
752
753
753
754
754
755
755
756
756
757
757
758
758
759
759
760
760
761
761
762
762
763
763
764
764
765
765
766
766
767
767
768
768
769
769
770
770
771
771
772
772
773
773
774
774
775
775
776
776
777
777
778
778
779
779
780
780
781
781
782
782
783
783
784
784
785
785
786
786
787
787
788
788
789
789
790
790
791
791
792
792
793
793
794
794
795
795
796
796
797
797
798
798
799
799
800
800
801
801
802
802
803
803
804
804
805
805
806
806
807
807
808
808
809
809
810
810
811
811
812
812
813
813
814
814
815
815
816
816
817
817
818
818
819
819
820
820
821
821
822
822
823
823
824
824
825
825
826
826
827
827
828
828
829
829
830
830
831
831
832
832
833
833
834
834
835
835
836
836
837
837
838
838
839
839
840
840
841
841
842
842
843
843
844
844
845
845
846
846
847
847
848
848
849
849
850
850
851
851
852
852
853
853
854
854
855
855
856
856
857
857
858
858
859
859
860
860
861
861
862
862
863
863
864
864
865
865
866
866
867
867
868
868
869
869
870
870
871
871
872
872
873
873
874
874
875
875
876
876
877
877
878
878
879
879
880
880
881
881
882
882
883
883
884
884
885
885
886
886
887
887
888
888
889
889
890
890
891
891
892
892
893
893
894
894
895
895
896
896
897
897
898
898
899
899
900
900
901
901
902
902
903
903
904
904
905
905
906
906
907
907
908
908
909
909
910
910
911
911
912
912
913
913
914
914
915
915
916
916
917
917
918
918
919
919
920
920
921
921
922
922
923
923
924
924
925
925
926
926
927
927
928
928
929
929
930
930
931
931
932
932
933
933
934
934
935
935
936
936
937
937
938
938
939
939
940
940
941
941
942
942
943
943
944
944
945
945
946
946
947
947
948
948
949
949
950
950
951
951
952
952
953
953
954
954
955
955
956
956
957
957
958
958
959
959
960
960
961
961
962
962
963
963
964
964
965
965
966
966
967
967
968
968
969
969
970
970
971
971
972
972
973
973
974
974
975
975
976
976
977
977
978
978
979
979
980
980
981
981
982
982
983
983
984
984
985
985
986
986
987
987
988
988
989
989
990
990
991
991
992
992
993
993
994
994
995
995
996
996
997
997
998
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375

```

```

37     digital_catalogs_iam_role_arn  =
  data.terraform_remote_state.nonprod_aws_infra.outputs.digital_bu_catalog_cloud_resources.lab
  _catalog.data_access_iam_role.arn
38     digital_ext_tbl_schema_role_arn =
  data.terraform_remote_state.nonprod_aws_infra.outputs.digital_bu_catalog_cloud_resources.lab
  _catalog.ext_tbl_data_access_iam_role.arn
39     digital_bu_lab = {
40       subnets      = ["subnet-039d49c832c613766", "subnet-04264bb265a9b477f", "subnet-
  06c48e0b4be6a9b1c"] //pre-defined
41       sgs        =
  data.terraform_remote_state.nonprod_aws_infra.outputs.digital_bu_lab_wks_cloud_resources["se
  curity_group_ids"]
42       root_bucket =
  data.terraform_remote_state.nonprod_aws_infra.outputs.digital_bu_lab_wks_cloud_resources["wo
  rkspace_root_bucket_name"]
43       xacc_role   =
  data.terraform_remote_state.nonprod_aws_infra.outputs.digital_bu_lab_wks_cloud_resources["cr
  oss_account_role_arn"]
44     }
45     digital_bu_field = {
46       subnets      = ["subnet-011451954924d667a", "subnet-057a5a1f9b2d4d983", "subnet-
  0d360b9a1fdb82ccb"] //pre-defined
47       sgs        =
  data.terraform_remote_state.nonprod_aws_infra.outputs.digital_bu_field_wks_cloud_resources["se
  curity_group_ids"]
48       root_bucket =
  data.terraform_remote_state.nonprod_aws_infra.outputs.digital_bu_field_wks_cloud_resources["wo
  rkspace_root_bucket_name"]
49       xacc_role   =
  data.terraform_remote_state.nonprod_aws_infra.outputs.digital_bu_field_wks_cloud_resources["cr
  oss_account_role_arn"]
50     }
51   }
52   prod = ...
53 }
54 }
```

Navigate to `.gitlab-ci.yml`, establish BU plan and apply as Terraform state files are separated based on Business Units. With `digital` BU as an example:

```

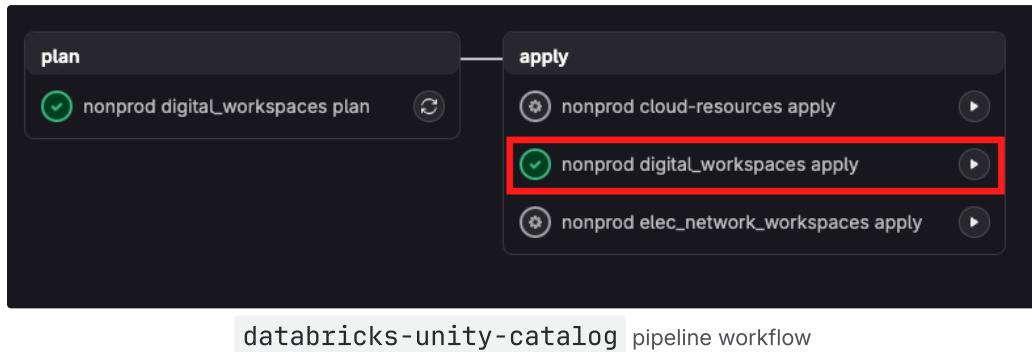
1 nonprod digital_workspaces plan:
2   stage: plan
3   # artifacts:
4   #   paths:
5   #     - stacks/**/.tfplan
6   variables:
7     STATEFILE_NAME: "digital-bu"
8     ENV: "nonprod"
9     DATABRICKS_HOST: "https://accounts.cloud.databricks.com"
10    DATABRICKS_CLIENT_ID: $app_datahub_nonprod_databricks_account_sp_clientid
11    DATABRICKS_CLIENT_SECRET: $app_datahub_nonprod_databricks_account_sp_secret
12    DATABRICKS_ACCOUNT_ID: $app_datahub_nonprod_databricks_account_id
13    TF_VAR_business_unit: "digital"
14    TF_VAR_client_id: $DATABRICKS_CLIENT_ID
15    TF_VAR_client_secret: $DATABRICKS_CLIENT_SECRET
```

```

16   TF_VAR_databricks_account_id: $DATABRICKS_ACCOUNT_ID
17   TF_VAR_aws_account_id: $nonprod_aws_account_id
18   # TF_VAR_target_vpc_id: $nonprod_vpc_id
19   TF_VAR_ci_job_token: $CI_JOB_TOKEN
20   before_script:
21     - *terraform-ver-init
22   script:
23     - cd stacks/digital-bu
24     - ../../scripts/tf-plan.sh
25   rules:
26     - changes:
27       - stacks/digital-bu/**/*
28       - modules/managed-workspace/**/*
29       - modules/rbac-hierarchy/**/*

```

Finally, on Gitlab repository, go to **Operate** (Column bar on the Left-hand side) > Under Business Unit Terraform State (**nonprod-digital-bu** for example) > click through the **Pipeline #** > click the BU text with image shown below.



Record the corresponding

- `workspace_url`,
- `ci_sp_client_id`, and
- depending on `prod` or `nonprod`, either record secret in `nonprod` or conduct data block `terraform_remote_state` call for `<lab/field>_ci_sp_oidc["secret"]` from prod

Establish the remaining `databricks.workspaces` modules by updating BU (in this case `stacks/digital-bu` and `stacks/elec-network-bu`) workspace host URL `providers.tf` below

```

1 locals {
2
3   # updating workspace host URL below
4   workspace_host_url = {
5     nonprod_lab    = "https://dbc-#####-.cloud.databricks.com"
6     nonprod_field = "https://dbc-#####-.cloud.databricks.com"
7     prod_lab       = "https://jemena-digital-lab.cloud.databricks.com"

```

```

8     prod_field    = "https://jemena-digital-field.cloud.databricks.com"
9   }
10 }
11
12 provider "databricks" {
13   alias      = "account"
14   host       = "https://accounts.cloud.databricks.com"
15   account_id = var.databricks_account_id
16 }
17
18 provider "databricks" {
19   //any workspace provider is fine, this is used for grants, see
20   //https://registry.terraform.io/providers/databricks/databricks/latest/docs/resources/grant
21   alias = "lab_mws"
22   host  = local.workspace_host_url["${var.environment}_lab"] //digital lab
23 }
24
25 provider "databricks" {
26   //any workspace provider is fine, this is used for grants, see
27   //https://registry.terraform.io/providers/databricks/databricks/latest/docs/resources/grant
28   alias = "field_mws"
29   host  = local.workspace_host_url["${var.environment}_field"] //digital field
30 }
31
32 provider "gitlab" {
33   alias = "databricks_group"
34   token = var.databricks_gitlab_token
35 }
36
37 provider "gitlab" {
38   alias = "digital_analytics_group"
39   token = var.digital_analytics_gitlab_token
40 }
```

and hence the remaining `catalog.tf` and `bu_grants.tf`

#### Gitlab Variable provisioning

To allow the following CI/CD variables to propagate across workspaces, please ensure you have the Gitlab repo group owner access.

- `<BU>_<field/lab>_wks_ci_<nonprod/prod>_sp_client_id`
- `<BU>_<field/lab>_wks_ci_<nonprod/prod>_sp_client_secret`

For checking Gitlab Group Access, navigate to **Subgroup and projects >** and check if you have `OWNER` tag on required subgroup (in this case **Databricks** and **Digital Analytics**)

if not, please contact the platform team representative to add you as a group owner

The screenshot shows the Future Networks Datahub dashboard. Key statistics include:

- Recent activity: Last 30 days (121)
- Merge requests created: 121
- Issues created: 0
- Members added: 1

Subgroups and projects listed:

- Databricks (Owner)
- Digital Analytics (Owner)

Then click through the group and navigate **Settings** (on the left-side bar tabs) > **Access Tokens** > **Add New Token** > Define **Token name** and **Select all scopes except ai\_features, k8\_proxy, manage\_runner, write\_registry** > Create group access token.

**ENSURE you copy and retain the access token until the next step as pop-up window will only appear once**

The screenshot shows the Jemena interface with the following navigation path: Jemena / Projects / Future Networks Datahub / Databricks / Access tokens.

**Group access tokens**

Generate group access tokens scoped to this group for your applications that need access HTTP(S). Learn more.

**Active group access tokens**

Token name	Description	Scopes
gitlab_provider_token	-	api, read_api, read_repository, write_repository, read_registry, write_registry
databricks-service-principal-propagator		api, read_api, read_repository, write_repository

Next, copy the Access Token into **databricks-unity-catalog** repo, to do so, navigate to [databricks-unity-catalog](#) > **Settings** > **Variables** > **Add variable** > Enter the the following attributes

Add Variable Attributes	Value
Visibilities	Masked and hidden

Flags	<Unselect all entries>
Descriptions	Derived from Gitlab Token < <b>Token Name</b> >
Key	<project name>_project_gitlab_token
Value	<Pasted value from previous steps>

Finally, utilise `databricks-unity-catalog` repo, navigate to `stacks/<business unit>-bu/provider.tf`, ensure the related project blocks (e.g. `databricks-groups`, and `digital-analytics` group above) are appended

```

1 provider "gitlab" {
2   alias = "databricks_group"
3   token = var.databricks_gitlab_token
4 }
5
6 provider "gitlab" {
7   alias = "digital_analytics_group"
8   token = var.digital_analytics_gitlab_token
9 }
```

and these variables are declared on `variables.tf` under related stack & CI variables declared under `.gitlab-ci.yml` for BU related pipelines below

```

1 <bussiness unit>-bu-<plan/apply>-nonprod:
2   stage: plan
3
4   ...
5   variables:
6     STATEFILE_NAME: "digital-bu"
7     ENV: "nonprod"
8     DATABRICKS_HOST: "https://accounts.cloud.databricks.com"
9     DATABRICKS_CLIENT_ID: $app_datahub_nonprod_databricks_account_sp_clientid
10    DATABRICKS_CLIENT_SECRET: $app_datahub_nonprod_databricks_account_sp_secret
11    DATABRICKS_ACCOUNT_ID: $app_datahub_nonprod_databricks_account_id
12
13    ...
14    TF_VAR_databricks_gitlab_token: $databricks_project_gitlab_token
15    TF_VAR_digital_analytics_gitlab_token: $digital_analytics_project_gitlab_token
```

## 5 Remaining Databricks Workspace-level Resources

On [Jemena / Projects / Future Networks Datahub / Databricks / databricks-workspaces](#), examining the CI/CD Group variables via

**Settings** (on the left-hand bar) > **CI/CD** > Scroll under Group variables to view the following request

- <BU>\_<field/lab>\_wks\_ci\_<nonprod/prod>\_sp\_client\_id
- <BU>\_<field/lab>\_wks\_ci\_<nonprod/prod>\_sp\_client\_secret
- <BU>\_<field/lab>\_wks\_url

with the following screenshot as example below:

CI/CD Variables </> 8			
Key ↑	Value	Environments	Actions
digital_field_wks_ci_nonprod_sp_client_id ⓘ digital field workspace non prod CI SP client id Masked	***** ⓘ	All (default) ⓘ	✍️ 🗑️
digital_field_wks_ci_nonprod_sp_client_secret ⓘ Masked	***** ⓘ	All (default) ⓘ	✍️ 🗑️
digital_field_wks_url ⓘ	***** ⓘ	All (default) ⓘ	✍️ 🗑️
digital_lab_wks_ci_nonprod_sp_client_id ⓘ digital lab workspace non prod CI SP client id Masked	***** ⓘ	All (default) ⓘ	✍️ 🗑️
digital_lab_wks_ci_nonprod_sp_client_secret ⓘ digital lab workspace non prod CI SP client secret Masked	***** ⓘ	All (default) ⓘ	✍️ 🗑️
digital_lab_wks_url ⓘ digital lab workspace URL	***** ⓘ	All (default) ⓘ	✍️ 🗑️

CI/CD variables for databricks-workspaces

Following implementing BU stacks (example is `./stacks/digital-bu` ), with `bu_<lab/field>_storage_mapper` created according to the source system aligned from the data product design for that specific BU (e.g. SAP hana would be landing in `bronze_sap_hana` ):

```

1 // mapper BU medallion buckets
2 locals {
3   bu_lab_catalog_storage_mapper = {
4     "landing"      = local.bu_lab_catalog_s3_buckets["landing"].name
5     "bronze_<source system>" = local.bu_lab_catalog_s3_buckets["bronze"].name //e.g.
6     bronze_sap_hana
7     "silver"       = local.bu_lab_catalog_s3_buckets["silver"].name
8     "gold"         = local.bu_lab_catalog_s3_buckets["gold"].name
9   }
10
11 bu_field_catalog_storage_mapper = {
12   "landing"      = local.bu_field_catalog_s3_buckets["landing"].name
13   "bronze_<source system>" = local.bu_field_catalog_s3_buckets["bronze"].name
14   "silver"       = local.bu_field_catalog_s3_buckets["silver"].name
15   "gold"         = local.bu_field_catalog_s3_buckets["gold"].name
16 }
17
18 bu_qa_catalog_storage_mapper = {
19   "landing"      = local.bu_qa_catalog_s3_buckets["landing"].name
20   "bronze_<source system>" = local.bu_qa_catalog_s3_buckets["bronze"].name
21   "silver"       = local.bu_qa_catalog_s3_buckets["silver"].name
22   "gold"         = local.bu_qa_catalog_s3_buckets["gold"].name
}
```

Follow by configuring additional `yaml` block under `gitlab-ci.yml`, with `digital_workspaces` as example below:

```

1 nonprod digital_workspaces plan:
2   stage: plan
3   # artifacts:
4   #   paths:
5   #     - stacks/*.tfplan
6   variables:
7     STATEFILE_NAME: "digital-bu-wks-res"
8     ENV: "nonprod"
9     TF_VAR_business_unit: "digital"
10    TF_VAR_lab_workspace_ci_sp_id: $digital_lab_wks_ci_nonprod_sp_client_id
11    TF_VAR_lab_workspace_ci_sp_secret: $digital_lab_wks_ci_nonprod_sp_client_secret
12    TF_VAR_lab_workspace_url: $digital_lab_wks_url
13    TF_VAR_field_workspace_ci_sp_id: $digital_field_wks_ci_nonprod_sp_client_id
14    TF_VAR_field_workspace_ci_sp_secret: $digital_field_wks_ci_nonprod_sp_client_secret
15    TF_VAR_field_workspace_url: $digital_field_wks_url
16    TF_VAR_ci_job_token: $CI_JOB_TOKEN
17   before_script:
18     - *terraform-ver-init
19   script:
20     - cd stacks/digital-bu
21     - ../../scripts/nslookup.sh #debug
22     - ../../scripts/tf-plan.sh

```

## 6 Populate Allowed JARs/Init Scripts to Workspace Catalogs

To allow Maven and Init Scripts access within the Workspace catalog and compute resources, navigate to

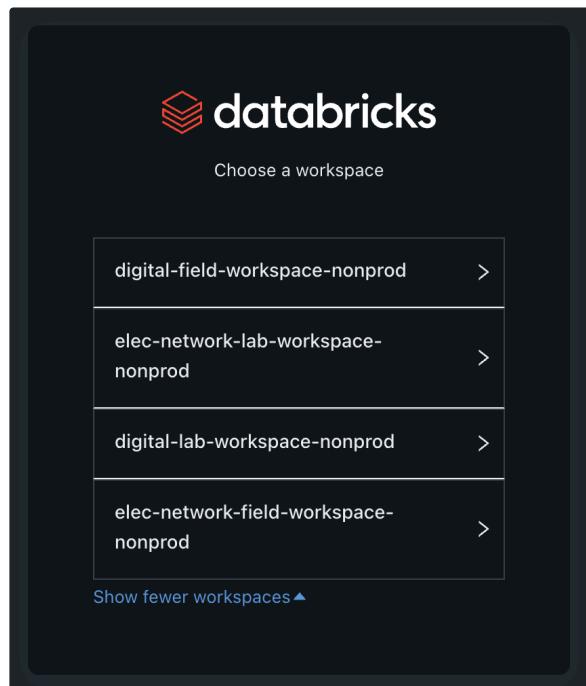
1. Click > > (or any metastore with naming conventions `app-datalab-<prod/nonprod>-jemena-metastore`)
2. Navigate to **Allowed JARs/Init Scripts** tab, and **add** for the following:

Type	Source Type	Source
Init script	Volume	/Volumes/digital_field_catalog/global/scripts
		/Volumes/digital_lab_catalog/global/scripts

		/Volumes/digital_qa_catalog/global/scripts
		/Volumes/elec_network_field_catalog/global/scripts
		/Volumes/elec_network_lab_catalog/global/scripts
		/Volumes/elec_network_qa_catalog/global/scripts
Maven	Coordinates	com.mysql:mysql-connector-j:9.0.0
		com.sap.cloud.db.jdbc:ngdbc:2.22.11
		com.sap.cloud.db.jdbc:ngdbc:2.22.12
		org.postgresql:postgresql:42.7.4

## Logging into Databricks workspaces

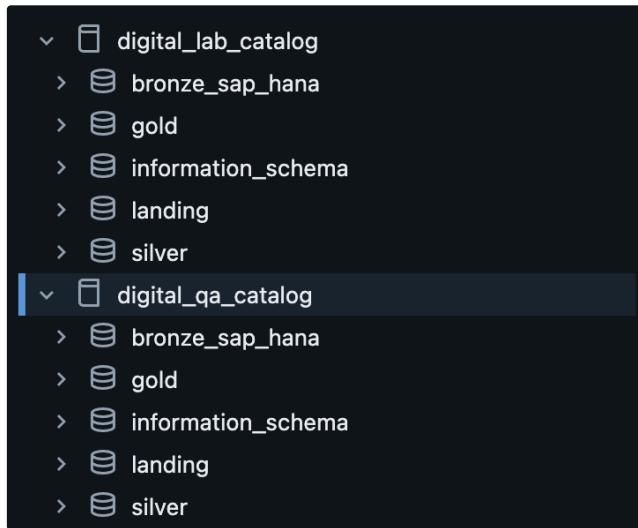
Using <https://accounts.cloud.databricks.com/> with your Jemena account (SSO sign-in), the available workspaces (subjected to RBAC) would be selectable like screenshot below:



Databricks Workspace Login Page

## Examining available Catalogs & associated backend for the given workspace

Whilst logging into workspace page, navigate to  icon via the column bar on your left-hand side. The available `lab` and `qa` catalogs and their managed tables for `digital-lab` workspaces (as an example can be seen as follow)



Then, click the Catalog of interests (`bronze_sap_hana` under `digital_lab_catalog` as an example) > then click **Details** located on the top bar under managed table name, you may see the following:

Name	bronze_sap_hana
Catalog Name	<code>digital_lab_catalog</code>
Owner	<code>bef2c7f1-af2a-431a-88bc-0142c04baf52</code>
Properties	managed_by=terraform purpose=Used for bronze_sap_hana
Storage Root	<a href="s3://app-datahub-nonprod-s3-digital-lab-bronze/data">s3://app-datahub-nonprod-s3-digital-lab-bronze/data</a>
Metastore Id	<code>9f5dba16-22ca-4d44-9cd2-b1dee875fd6a</code>
Created At	11/10/2024, 9:56:38 pm
Created By	<code>bef2c7f1-af2a-431a-88bc-0142c04baf52</code>
Updated At	11/10/2024, 9:56:38 pm
Updated By	<code>bef2c7f1-af2a-431a-88bc-0142c04baf52</code>
Catalog Type	MANAGED_CATALOG
Storage Location	<a href="s3://app-datahub-nonprod-s3-digital-lab-bronze/data/_unitystorage/schemas/1242e3b7-5cbe-4856-b2b9-c2f10c3d1bd3">s3://app-datahub-nonprod-s3-digital-lab-bronze/data/_unitystorage/schemas/1242e3b7-5cbe-4856-b2b9-c2f10c3d1bd3</a>
Schema Id	<code>1242e3b7-5cbe-4856-b2b9-c2f10c3d1bd3</code>
Browse Only	false

All tables and the associated S3 backend (i.e. `S3://...` the link next to `Storage Root`) and exhibits MANAGED TABLE type (i.e. `MANAGED_CATALOG` type).

