Unit 5 – Developing with SAP Extension Suite

Certification: C_CPE_13

Unit 5 – Manual Deployment

Agenda

- What have we learned so far
- Manual Deployment using manifest.yaml file (CF CLI)
- Manual Deployment using mta.yaml file
- Summary points

Steps involved

1. Initialize full-stack project – Completed (Unit 1, 2)
2. Create the tables – Data Modeling – Completed (Unit 1, 2)
3. Generic handlers – Out-of-the-box CRUD functionality – Completed (Unit 1, 2)
4. Basic UI – Completed (Unit 3)
5. List Report layout – Completed (Unit 3)
6. Custom event handling – Business logic – Completed (Unit 3)
7. Support for external API – Completed (Unit 4)
8. Connecting to Sandbox – Completed (Unit 4)

Steps involved

9. Consume external service in UI – Completed (Unit 4)
10. Manual deployment to CF using manifest.yml
11. Manual deployment to CF using mta.yml
12. Security – Restrictions and Roles
13. Security – Authorization and Trust Management
14. Creating an AppRouter
15. Adding AppRouter to mta
16. CI / CD Pipeline

Step 10 – Manual deployment (manifest.yml)

```
git checkout 9_manual_deploy_using_manifest (Use tab for branch name)
```

New Files

- xs-security.json
- manifest.yml

Modified Files

- package.json

Deploy command

- cf push

Deployment Steps – manifest.yml (CF CLI)

Steps involved:

1. Create HDI container on SAP HANA Cloud
2. Update project to use SAP HANA (for Production)
3. Create Destination Service
4. Create Authorization & Trust Management Service
5. Create and update manifest.yml file
6. Update package.json file
7. Deploy to CF

Deployment Steps – manifest.yml (CF CLI)

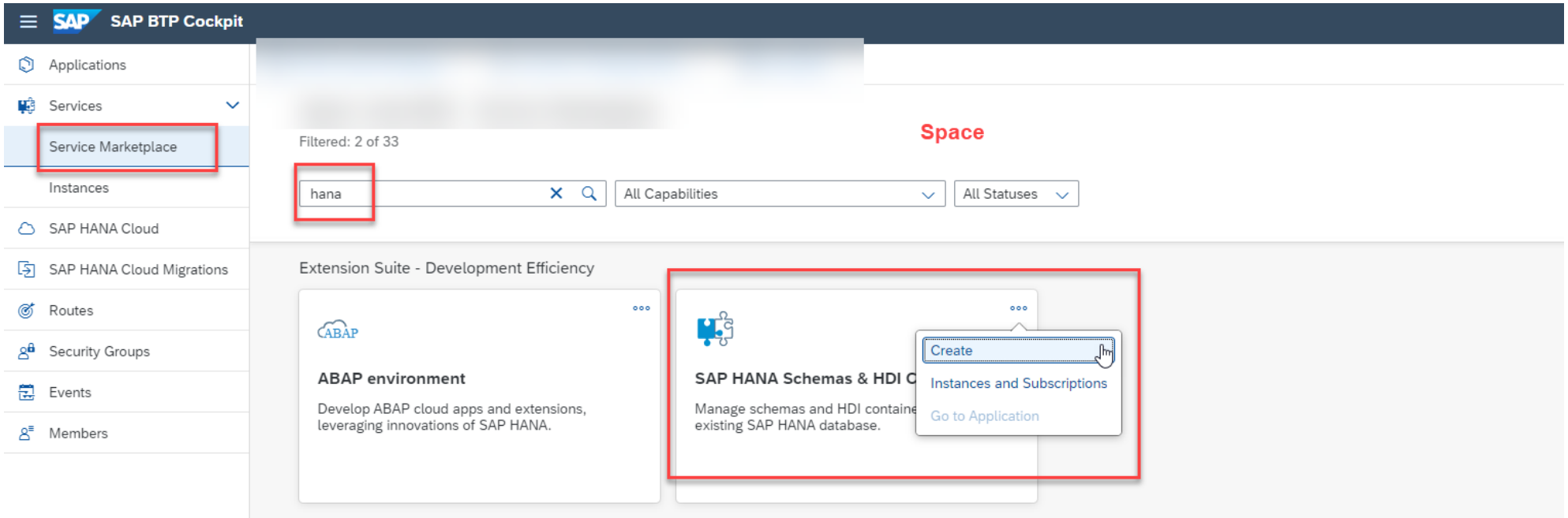
Create external services in Cloud Foundry

- Command line interface

  cf create-service <service_name> <service_plan> <name_of_service_instance>

  cf create-service hana hdi-shared myhana-db

- SAP BTP Cockpit



Deployment Steps – manifest.yml (CF CLI)

Create HDI container on SAP HANA Cloud

- `cf create-service hana hdi-shared risk-management-db`

Update project to use SAP HANA (for Production)

- `cds add hana`

Create Destination Service

- `Sandbox API is publicly available.  No need for Connectivity Service`
- cf create-service destination lite risk-management-destination-service

Create Authorization & Trust Management Service

- `Create xs-security.json file`
- `cf create-service xsuaa application risk-management-xsuaa –c xs-security.json`

Deployment Steps – manifest.yml (CF CLI)

Create and update manifest.yml

- `cds add cf-manifest`

Update package.json

- `Update package.json`
- `cds build --production`

Deploy to CF

- `cf push`

- `cf set-env risk-management-srv apikey JIzPB8YwC3gF----------------`
- `cf restart risk-management-srv`

Step 11 – Manual deployment (mta.yml)

`git checkout 10_manual_deploy_using_mta (Use tab for branch name)`

New Files
- `mta.yaml`

Modified Files
- `package.json`

Deploy command
- `mbt build`
- `cf deploy mta_archive/ risk-management_1.0.0.mtar`

Deployment Steps – mta.yml

Create and update mta.yml

- `cds add mta`

Update package.json

- `Update package.json`
- `cds build --production`

Deploy to CF

- `mbt build –t ./`
- `cf deploy risk-management_1.0.0.mtar`

MTA Deployment Descriptor

Module

- self-contained application – to be developed, packaged, deployed

Module Type

- type of application (html5, nodejs)

Resource

- external service required by the module

Property

- key-value pair of module, resource used during deployment or runtime

Parameter

- reserved variable belonging to module or resource used during deployment or runtime

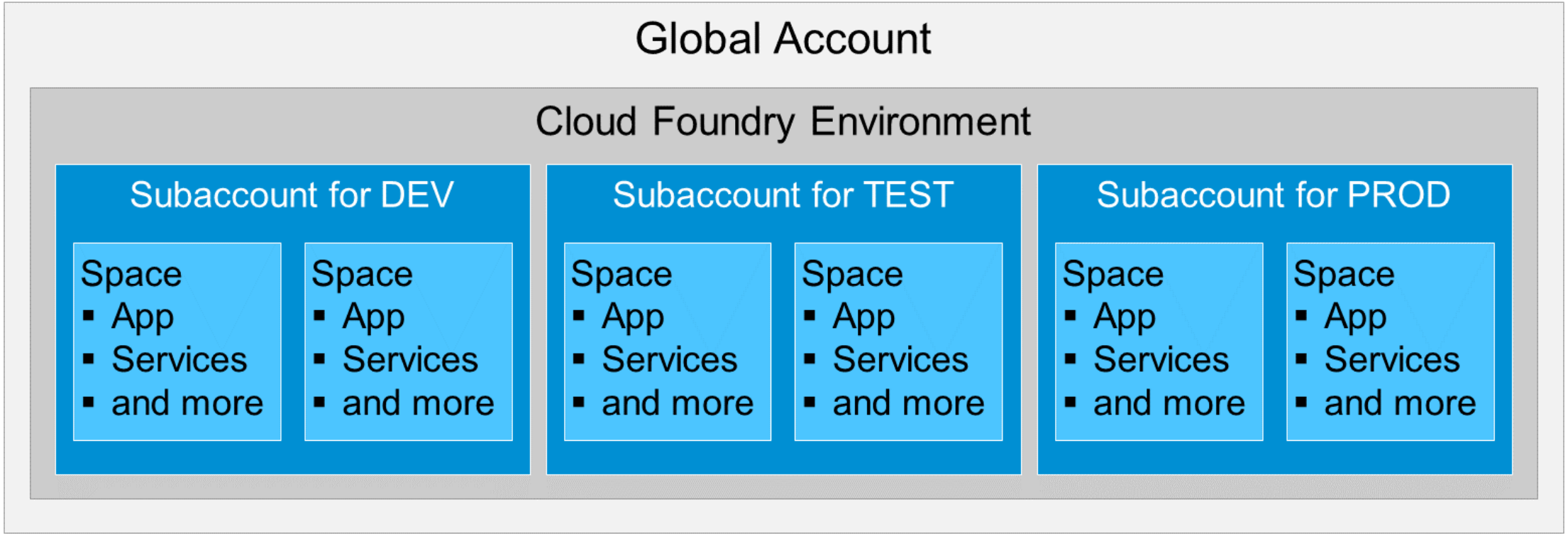
Dependency

- provides – properties or parameters provided by module to other modules
- requires – modules or resources required by a module to run

MTA Deployment Descriptor

```yaml
- name: risk-management-srv  # OData Service module
  type: nodejs  # Type of module - nodejs
  path: gen/srv
  parameters:
    buildpack: nodejs_buildpack  # buildpack is a reserved variable used during deployment
  requires:
   # External service required for this module to run...
   - name: risk-management-db
   - name: risk-management-xsuaa
   - name: risk-management-destination-service
  provides:
   - name: srv-api      # required by consumers of CAP services (e.g. approuter)
     properties:
       srv-url: ${default-url}
```

Cloud Foundry Environment



When you create a Subaccount and enable CF runtime – System automatically creates a Cloud Foundry org
Subaccount and org have a 1:1 relationship

You can create multiple spaces within Cloud Foundry org
In Cloud Foundry environment, you deploy applications and consume services at the space level

Cloud Foundry Environment

Manage Cloud Foundry environment

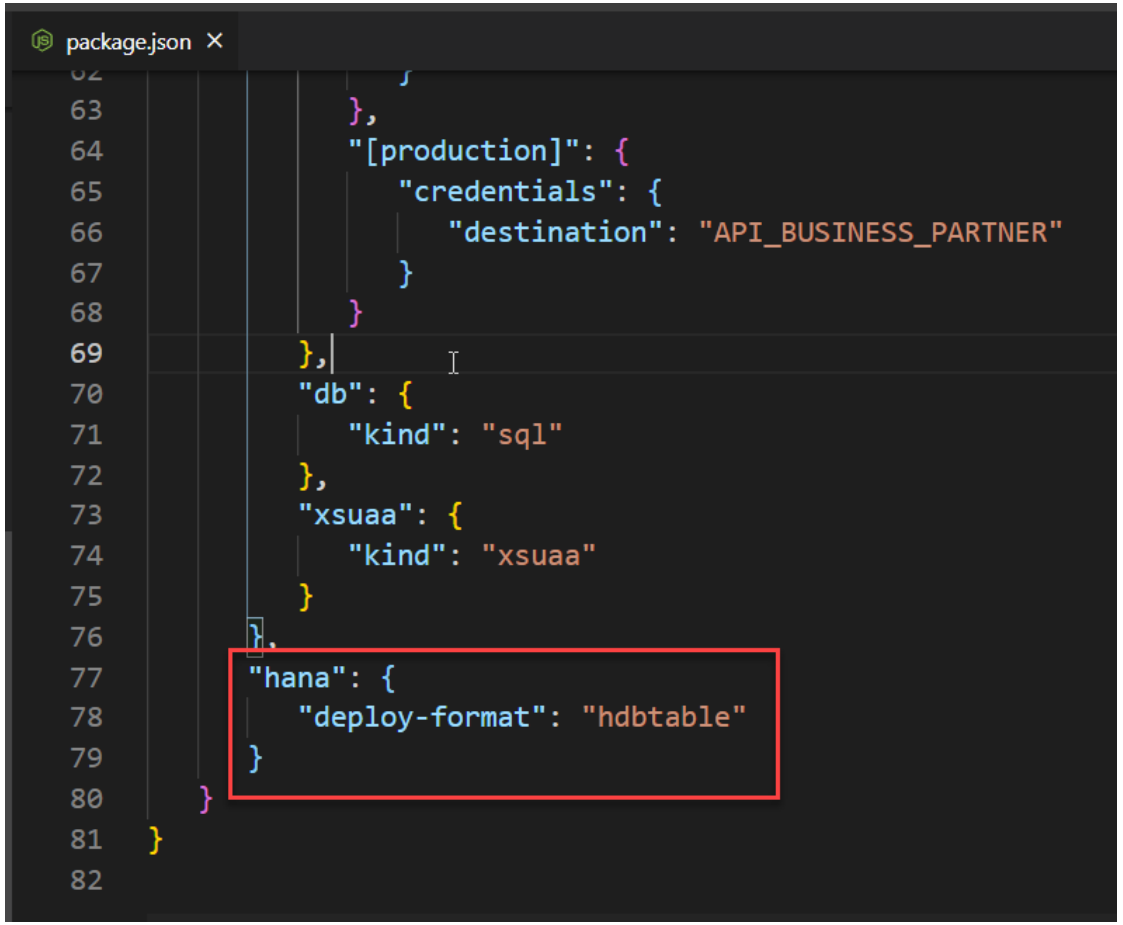
- SAP BTP Cockpit
- CF CLI (more powerful)

Key Summary Points – Unit 5

`cds add hana`

Modified file:

- package.json



Key Summary Points – Unit 5

Question 2

Choose the correct answer(s).

What are advantages of using an MTA file for deployment? (Choose 2)

- [ ] It supports red - green deployment.
- [x] It supports blue-green deployment.
- [ ] It provides workflows.
- [x] It provides a build tool.

Feedback

Correct. The advantages of using an MTA file for deployment are: "it supports blue-green deployment", and "it provides a build tool".

Key Summary Points – Unit 5

MTA File overview

Name	Description
mta.yaml	**Development** descriptor for a multi-target application (MTA). The information in the mta.yaml file provides instructions for the MTA development and build process.
mtad.yaml	**Deployment** descriptor for a multi-target application (MTA). The information in the mtad.yaml file provides instructions for the deploy service.
mtaext.yaml	**Deployment extension descriptor** (optional). This is used to provide system-specific details that are not known until deployment time.

Key Summary Points – Unit 5

Question 4

Choose the correct answer(s).

Which statements about YAML files are correct? (Choose 2)

- [x] YAML uses whitespace indentation for structuring purposes.
- [ ] YAML uses tab indentation for structuring purposes.
- [ ] YAML uses hyphens: - for comments.
- [x] YAML uses hashes: # for comments.

Feedback

Correct. The following statements are correct: "YAML uses whitespace indentation for structuring purposes", and "YAML uses hashes: # for comments".

Key Summary Points – Unit 5

Question 5

Choose the correct answer.

Which concept describes Cloud Foundry applications?

- ○ Monoglot
- ✔ ● Polyglot
- ○ Proglot
- ○ Epiglot

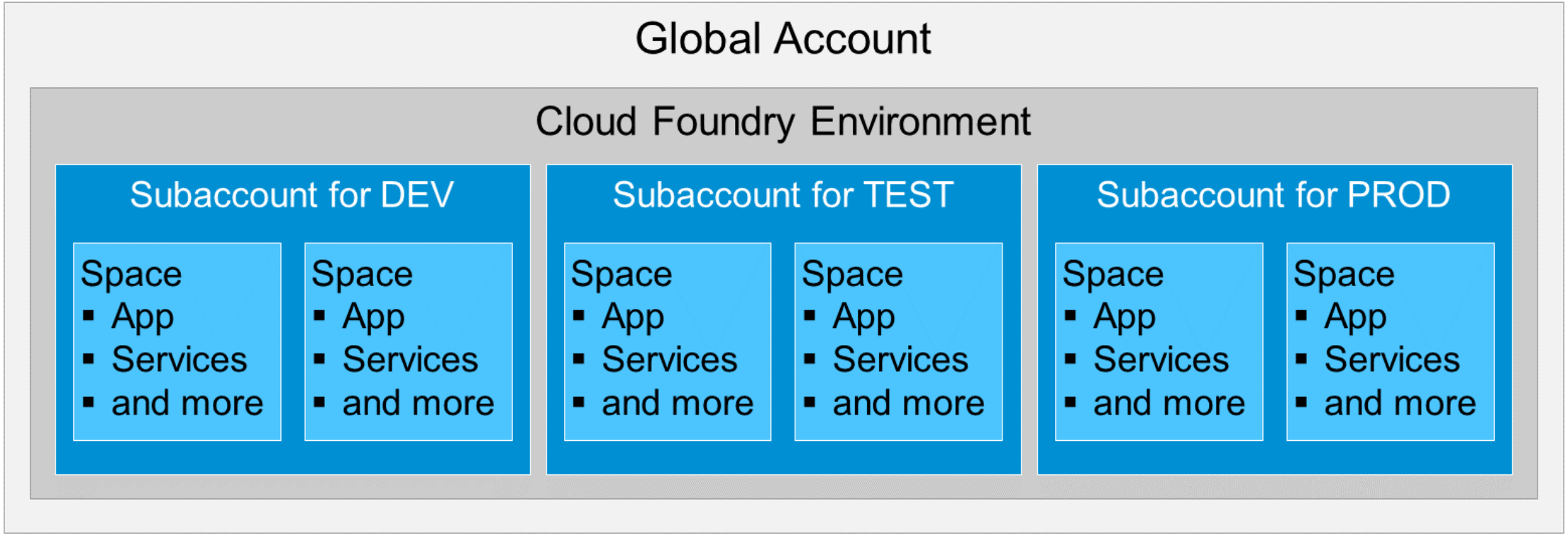
Feedback

Correct. The concept "polyglot" describes Cloud Foundry applications.

Cloud Foundry includes a set of system buildpacks for common languages and frameworks. This table lists the system buildpacks.

Name	Supported Languages, Frameworks, and Technologies	GitHub Repository
Binary	*n/a*	Binary source
Go	Go	Go source
HWC	HWC	HWC source
Java	Grails, Play, Spring, or any other JVM-based language or framework	Java source
.NET Core	.NET Core	.NET Core source
NGINX	NGINX	NGINX source
Node.js	Node or JavaScript	Node.js source
PHP	Cake, Symfony, Zend, NGINX, or HTTPD	PHP source
Python	Django or Flask	Python source
R	R	R source
Ruby	Ruby, JRuby, Rack, Rails, or Sinatra	Ruby source
Staticfile	HTML, CSS, JavaScript, or NGINX	Staticfile source

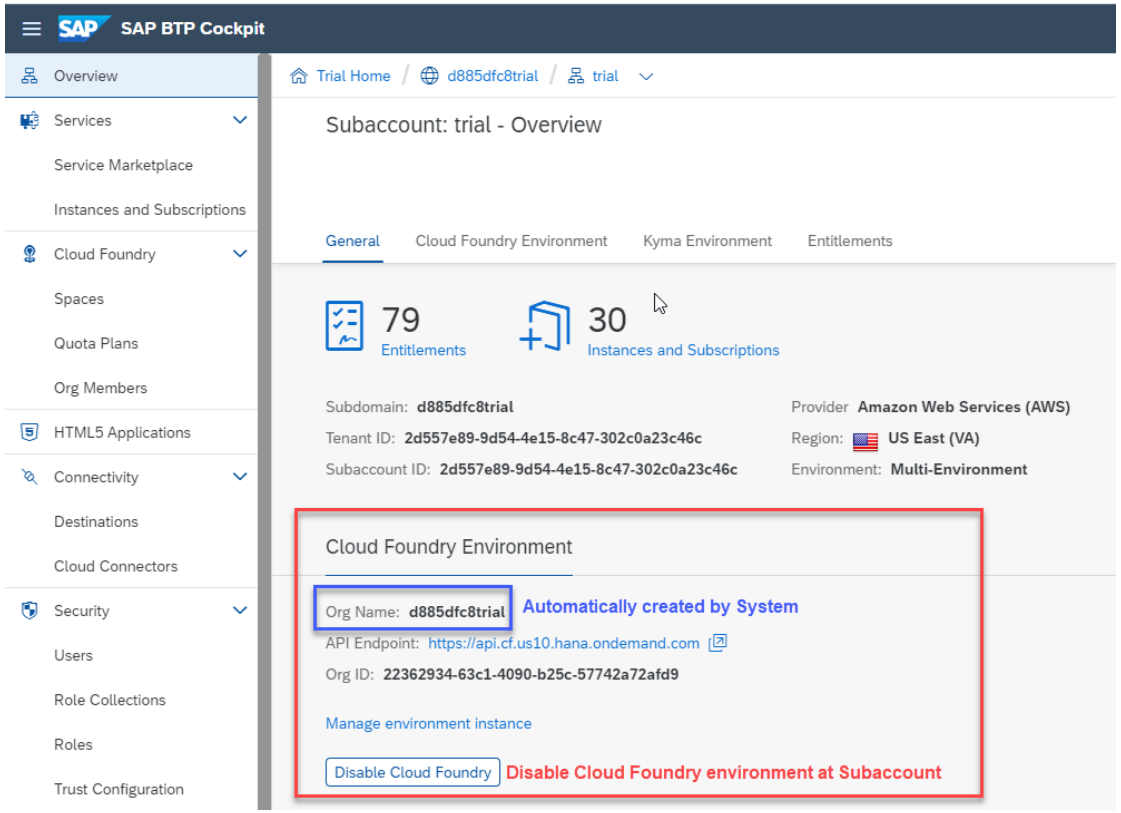
Key Summary Points – Unit 5

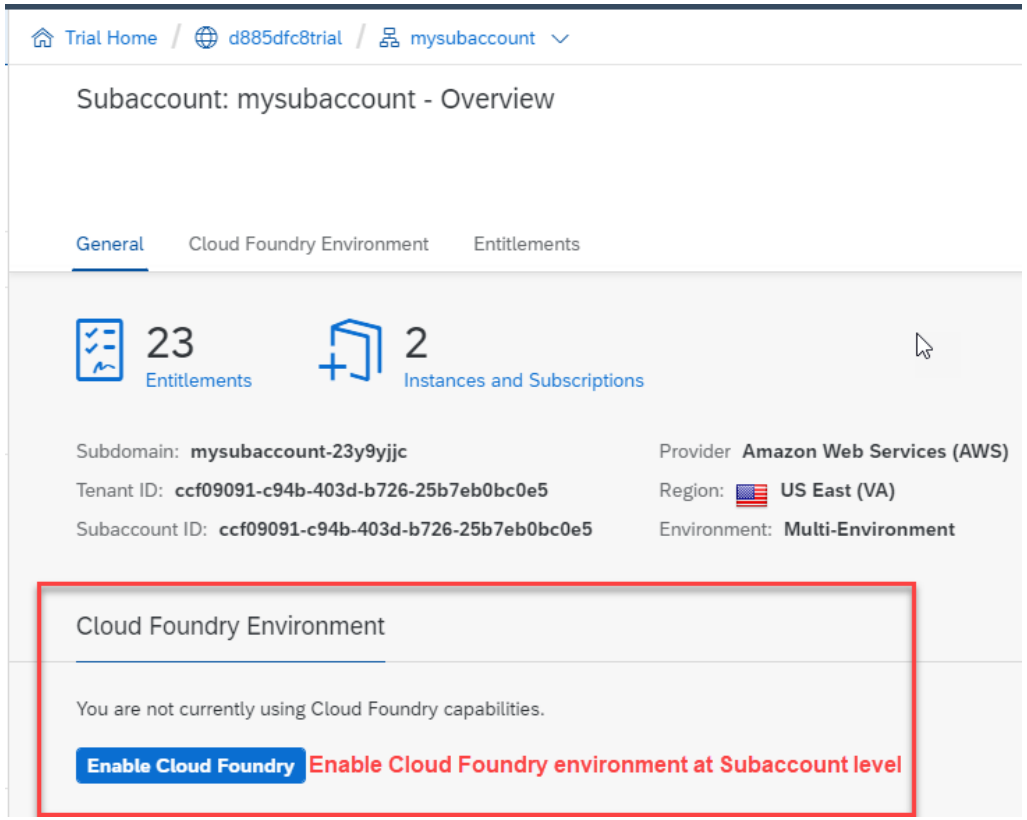


When you create a Subaccount and enable CF runtime – System automatically creates a Cloud Foundry org
Subaccount and org have a 1:1 relationship

You can create multiple spaces within Cloud Foundry org
In Cloud Foundry environment, you deploy applications and consume services at the space level

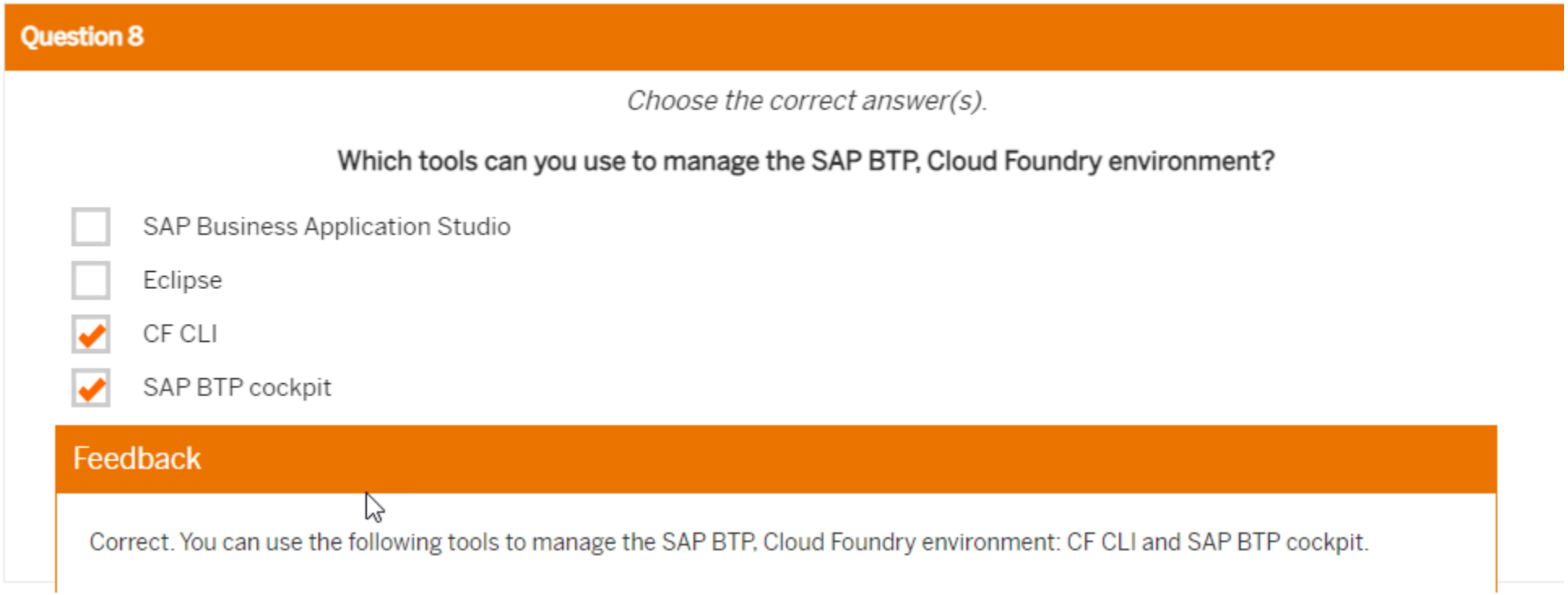
Key Summary Points – Unit 5





CF Org is automatically created by the system – when you enable Cloud Foundry environment
Subaccount and Org have a 1:1 relationship

Key Summary Points – Unit 5



Key Summary Points – Unit 5

SAP BTP-Specific Configurations

The following technical configurations are specific to SAP BTP and differ from the default configuration:

- By default, a newly pushed (or started) Cloud Foundry application needs to respond to a health check within the first 60 seconds, otherwise the application is considered to have failed. For more information, see https://docs.cloudfoundry.org/devguide/deploy-apps/healthchecks.html#health_check_timeout . On SAP BTP, however, you can override this timeout to up to 10 minutes. For instructions, see https://docs.cloudfoundry.org/devguide/deploy-apps/large-app-deploy.html .
- On SAP BTP, application SSH access is disabled by default. For more information on SSH, see https://docs.cloudfoundry.org/devguide/deploy-apps/app-ssh-overview.html .
- On SAP BTP, the Cloud Foundry API is protected by a rate limit against misuse. The limit is in the range of a few 10k requests per hour per user.
- In the Cloud Foundry environment, there's a logging rate limit to guard against malicious applications. The limit is in the range of up to a few thousand logs per second per application instance. If this limit is exceeded, additional logs from the application instance are dropped and a warning message is injected into the application instance's log stream every second. This message also contains the exact log rate limit.
- In the Cloud Foundry environment, applications get a guaranteed CPU share of ¼ core per GB instance memory. As the maximum instance memory per application is 8 GB, this allows for vertical scaling up to 2 CPUs.

  If applications running on the same virtual machine don't use their guaranteed CPU, other applications might get more CPU. This isn't guaranteed and might be subject to change in the future. If you encounter performance problems, scale up your application or increase the application start timeout.

  The number of running threads per application instance is limited to 10 420. Reaching this limit can cause performance issues.
- When pushing or scaling your application, you can define a `disk_quota` that can be up to 4 GB. For more information, see https://docs.cloudfoundry.org/devguide/deploy-apps/manifest-attributes.html#disk-quota .
- When deploying applications on SAP BTP, the maximum application package size is 1.5 GB. If your application is larger than that, the deployment fails. For more information, see https://docs.cloudfoundry.org/devguide/deploy-apps/large-app-deploy.html
- In global accounts that support the consumption-based commercial model you might see a quota limit for certain services. This is a technical limit only, not a business limit. If you need to increase this limit, report an incident to SAP support for component BC-NEO-CIS.
- In the Cloud Foundry environment, the SAP HANA database supports up to 1,000 simultaneous connections per database.
- Cloud Foundry Audit Events have a retention period of 14 days. For more information on Audit Events, see https://docs.cloudfoundry.org/running/managing-cf/audit-events.html .