**UNIT 3**

**Cloud Platform**

### Lesson 1

[Introducing Cloud Platform](#_bookmark0) 67

Lesson 2

[Explaining the Commercial Model for SAP Cloud Platform](#_bookmark1)

[Exercise 3: Show Services at discover-center.cloud](#_bookmark2)

72

77

### Lesson 3

[Explaining the Runtime of Cloud Foundry](#_bookmark3)

[Exercise 4: Show aSubaccount with Cloud Foundry](#_bookmark4)

80

87

Lesson 4

[Exploring the Service: ABAP](#_bookmark5) 93

### Lesson 5

[Explaining KYMA](#_bookmark6) 107

UNIT OBJECTIVES

* Introduce the basic components of SAP Cloud Platform
* Explain the Commercial Model of SAP Cloud Platform
* Explain the account model in Cloud Foundry
* Explain the difference between platform and business users
* Explore the most important components of the ABAP installation
* Explain, what Kyma is, what components Kyma is made of and what you can do with it

Unit 3

Lesson 1

# Introducing Cloud Platform

LESSON OBJECTIVES

After completing this lesson, you will be able to:

* Introduce the basic components of SAP Cloud Platform

### Basic Components of SAP Cloud Platform

In particular, we'll look at the following topics in this lesson:

* SAP Cloud Platform in a nutshell.
* SAP Cloud Platform Cockpit.
* The Environments
* Which services are available?
* SAP CP Reference Architecture.

SAP Cloud Platform in a nutshell

SAP Cloud Platform is an enterprise platform-as-a-service (enterprise PaaS) that provides:

* Comprehensive application development services and capabilities.
* Which lets you build, extend, and integrate business applications in the cloud.

SAP Cloud Platform offers two types of global accounts, Trial accounts and enterprise accounts. In the following we will only deal with the enterprise accounts.

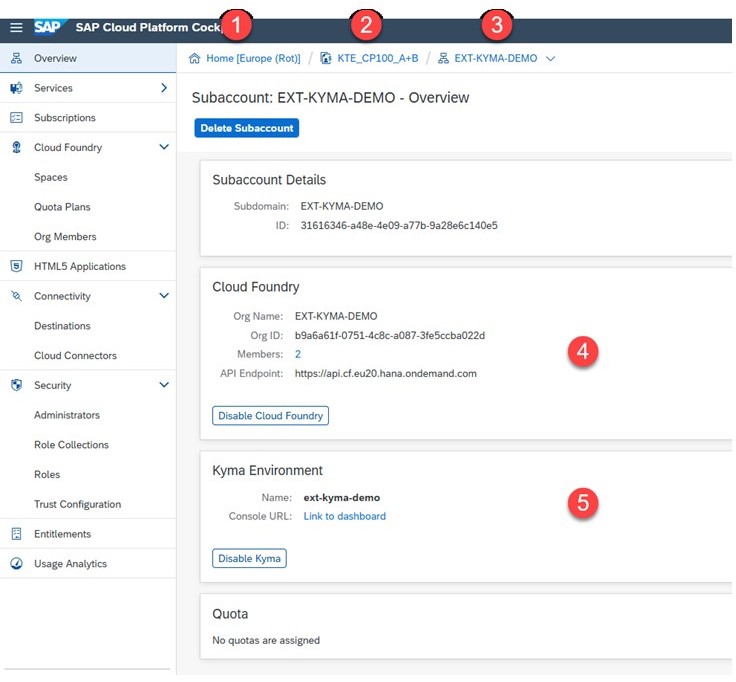
SAP Cloud Platform Cockpit



Figure 30: SAP Cloud Platform Cockpit of an Enterprise Account

On the screenshot you will see the following:

1. Region of the global account - here Europe Rot.
2. Global Account - here KTE\_CP100\_A+B.
3. Subaccount - here EXT-KYMA-DEMO.
4. Environment Cloud Foundry with organisation and endpoint - here: https:// api.cf.eu20.hana.ondemand.com (Netherlands Europa).
5. Environment Kyma with name ext-kyma-demo.

The Environments

Currently, the following environments are available:

* Cloud Foundry ( fully managed by SAP).
* ABAP ( platform as a Service).
* Kyma (open Build-on approach).
* More to come.

Cloud Foundry

SAP Cloud Platform Cloud Foundry environment is an open Platform-as-a-Service (PaaS) targeted at microservice development and orchestration.

SAP Cloud Foundry should be used if you prefer a Managed Build-on approach. Cloud Foundry is fully managed by SAP and you benefit from a high level of abstraction from the underlying infrastructure.

You don't have to deal with technical aspects such as virtual machines, networking, monitoring, and more. It provides out-ofthe-box integration into all mandatory kernel services.

Cloud Foundry runs on Kubernates (K8s)

ABAP

The ABAP environment is a platform as a service that allows you to extend existing ABAP-based applications and develop ABAP cloud apps decoupled from the digital core. You can leverage your ABAP know-how in the cloud and reuse existing ABAP assets by writing your source code with ABAP Development Tools for Eclipse.

Kyma

The Kyma environment ( Isteio with K8s) allows you to extend existing SAP systems with your own Functions or microservices.

Kyma is a platform for extending applications with serverless functions and microservices. It provides a selection of cloud-native projects glued together to simplify the creation and management of extensions.

Kyma should be used for an Open Build-on approach. It provides you with more flexibility by using containers and Kubernetes, to build cloud native applications. By using Kubernetes, it will be easier to develop a solution that is highly scalable.

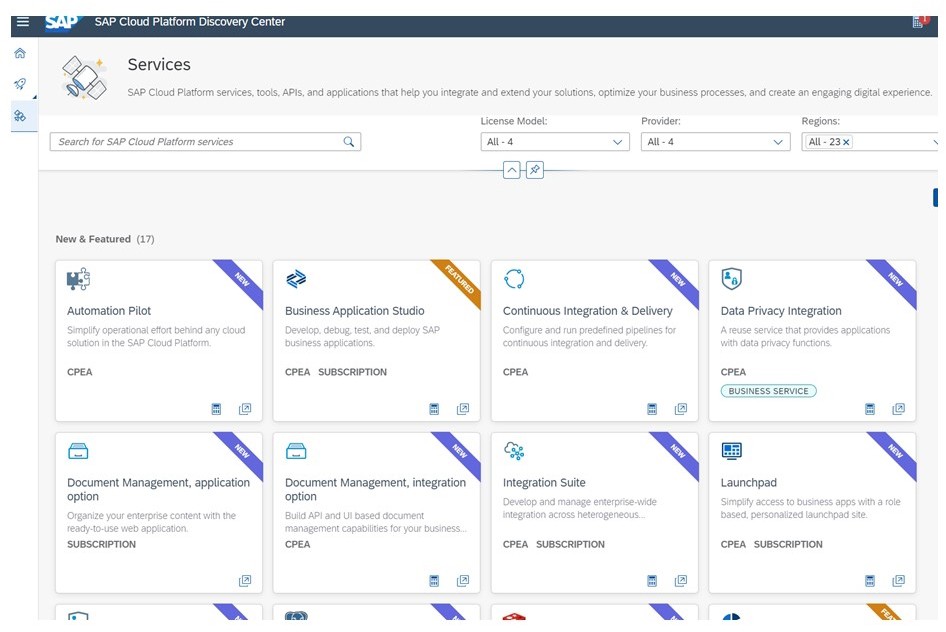
Which services are available?



Figure 31: Screenshot of the Discovery Center

All services currently available can be found at https://discovery-center.cloud.sap/ viewServices. There they are listed in the derivative of the regions, commercial models and more.

Not all Services are available at KYMA environment.



Note:

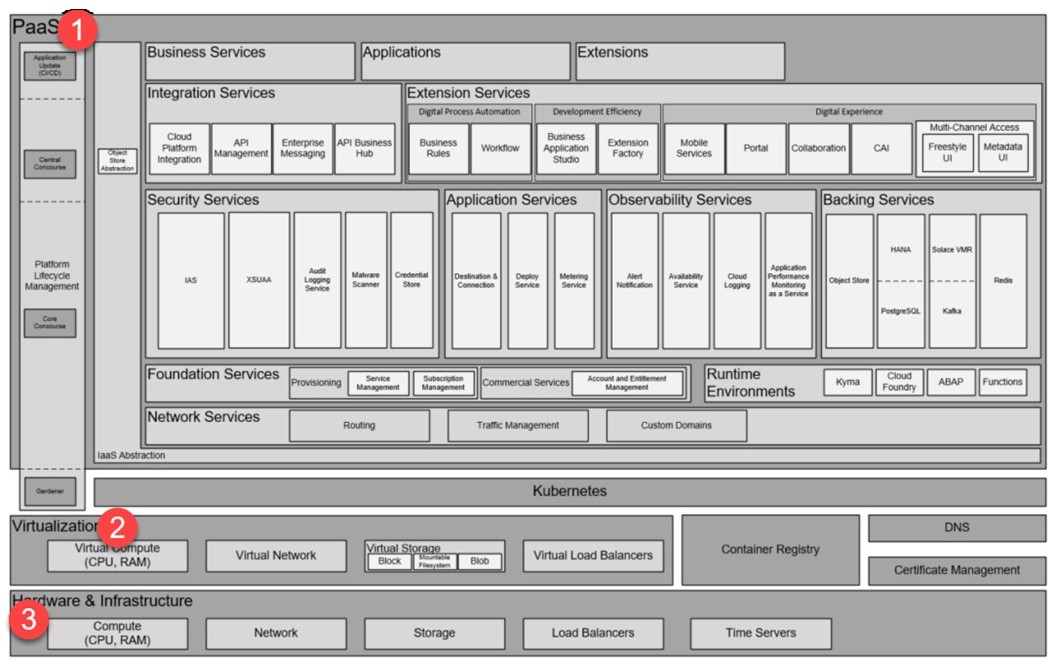
ABAP is also a service and not an environment.

SAP Cloud Platform Reference Architecture

So that you get an approximate overview of what the SAP Cloud Platform is from a technical point of view.



Figure 32: Overview of the SAP Cloud Foundry Reference Architecture

Explanations:

1. The PaaS Layer provides all functionality, applications and services we can use direct or indirect. It runs on K8s.
2. The Virtualisation Layer.
3. Hardware and Infrastructure (IaaS).

Get more in Detail

SAP sets the following priorities:

* Security
* Availability
* Operational Efficiency
* Performance
* Development Efficiency

Security

Customers, especially in managed cloud services scenarios, expect maximum security and data protection compliance when moving their business processes to the Cloud. SAP CP architectures are and should be built in a way that SAP's Product Standard Security is fulfilled.

Availability

Customers also expect an "always-on" behavior for Cloud services. Therefore, SAP CP targets a 99,9% availability

Operational Efficiency

Optimizing operations isn't just about cost optimization, but about empowering people to safely operate complex distributed systems

Performance

All SAP CP services and solutions should define relevant performance KPIs, and constantly measure them, optimize them and keep them equal or better than the target values.

Development Efficiency

Hyperscalers do not list development efficiency as a quality, but for SAP CP it is an important topic.

LESSON SUMMARY

You should now be able to:

* + Introduce the basic components of SAP Cloud Platform

Unit 3

Lesson 2

# Explaining the Commercial Model for SAP

Cloud Platform

LESSON OBJECTIVES

After completing this lesson, you will be able to:

* Explain the Commercial Model of SAP Cloud Platform

### The Commercial model of SAP Cloud Platform

In particular, we'll look at the following topics in this lesson:

* The nature of Services
* Applications
* SAP Cloud Platform – Choice of Commercial Models
* Structure of the Commercial Models
* Consumption-Based Commercial Model - CPEA

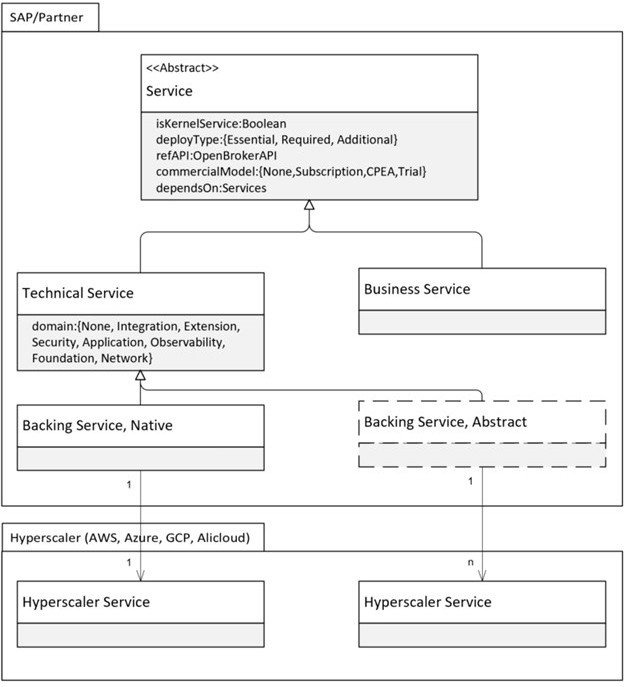
The nature of Services

Services constitute SAP CP as PaaS offering since major function blocks are implemented and exposed as service. Services can be accessed and managed via the Service Manager

Services have a provider and a consumer "view". For service providers it's an entity to "ship" functionality, manage the lifecycle and an operation unit. Consumers see it as re-use entity, for which an entitlement, configuration and access points (API and/or UI) are given. The default model for consuming services today is to separate service consumption from service provisioning completely. From a consumer perspective, this is ideal because the service is "provided as a Service" often-times with an SLA.



Figure 33: Provider View of a Service

Description of the figure:

* + A service is a Technical Service or a Business Service
  + Backing Services are either abstracted and exposed as Technical Service or it is natively consumed without any abstraction except the access via Service Manager (e.g. ObjectStore, MS Embrace project with OSB implementation)
  + A service can be a Kernel Service
  + A service has a deploy type. Essential services are deployed and exposed in all SAP CP data centers or availability zones respectively. Required services are deployed with the essential services since the essential services depend on them. Additional services are deployed on selected data centers or availability zones as needed on request. SAP CP offers approximately 70 essential or required technical services today, which are deployed on all SAP CP data centers and availability zones as defined in the Service Availabilitymatrix at the discovery center.

Applications

Applications building on or integrating with SAP Cloud Platform can use the Service Discovery and Management Kernel Service (22), also known as Service Manager, for the provisioning/ deprovisioning of service instances listed in the marketplace and creating/accessing/ updating/deleting credentials for these service instances.

The actual implementations of these operations are service specific, and services implement them in service brokers that they register with the Service Manager.

SAP Cloud Platform - Choice of Commercial Models Basically, there are two models:

* + The Subscription model licenses only the services you need.
  + In the Consumption model you can use all services but only pay for the use.

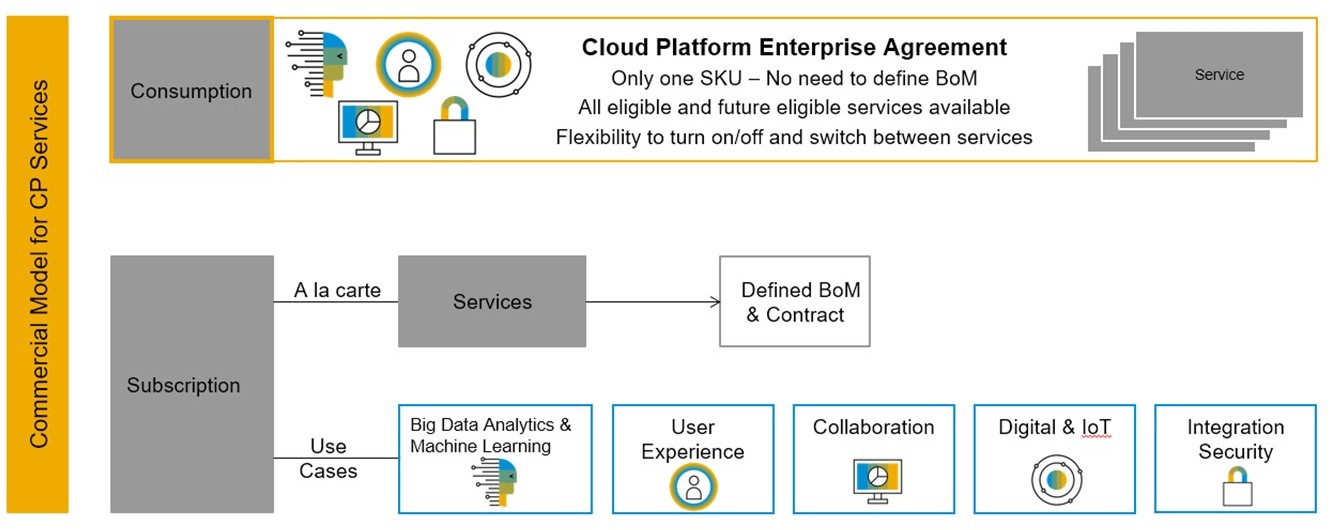


Figure 34: Overview Slide

Explanations about the slide:

Subscription

* You plan and pay in advance for every service separately (high touch),whether you use it or not.
* Certain contract duration (three month or more, typical three years!).
* Coarse granularity for capacity (blocks of for example 100 users, 5.000 visits). Consumption-based
* You get access to all eligible services, without quotas or limitations.
* Self-service activation and de-activation ("low touch").
* Only pay for what you use. Cloud Credits
* Cloud credits are a pre-paid commitment for the consumption of SAP Cloud Services in a defined time period à unused cloud credits expire at the end of the phase and contract year.
* Cloud credits are subject to discount.
* Purchasing of top-up cloud credits at any point of time.

Structure of the Commercial Models

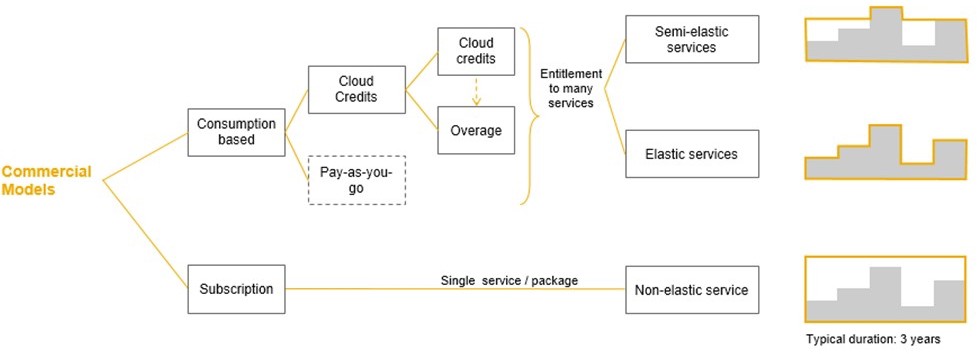


Figure 35: Structure of the Commercial Models

As you can see on the slide, the Consumption based Model is much more flexible. It offers Elastic services. The picture on the far right describes the ability of the model to adapt to load changes. The middle picture shows a completely flexible adjustment of the costs to the actual usage.

Consumption-Based Commercial Model - CPEA

CPEA (Cloud Platform Enterprise Agreement) is a Consumption-Based Commercial Model for SAP Cloud Platform.

Facts about Consumption-Based Commercial Model - CPEA:

* + You get access to all eligible services, without quotas or limitations.
  + Self-service activation and de-activation (“low touch”)
  + Only pay for what you use.

Go deeper:

The consumption-based model decouples:

* + Commercial agreements and
  + Technical adoption

to help customers easily to:

* + Find,
  + Try,
  + Buy,
  + Consume and
  + Pay

Facts about Cloud Credits:

* + Cloud credits are a pre-paid commitment for the consumption of SAP Cloud Services in a defined time period à unused cloud credits expire at the end of the phase and contract year.
  + Cloud credits are subject to discount.
  + Purchasing of top-up cloud credits at any point of time.
* Cloud credits entitle you to flexible usage of all consumption-based services in the portfolio, as well as future (yet to be introduced) services.
* Service usage is "debited" from the cloud credits.
* Excess usage ("overage") is invoiced.

Overview - Price List

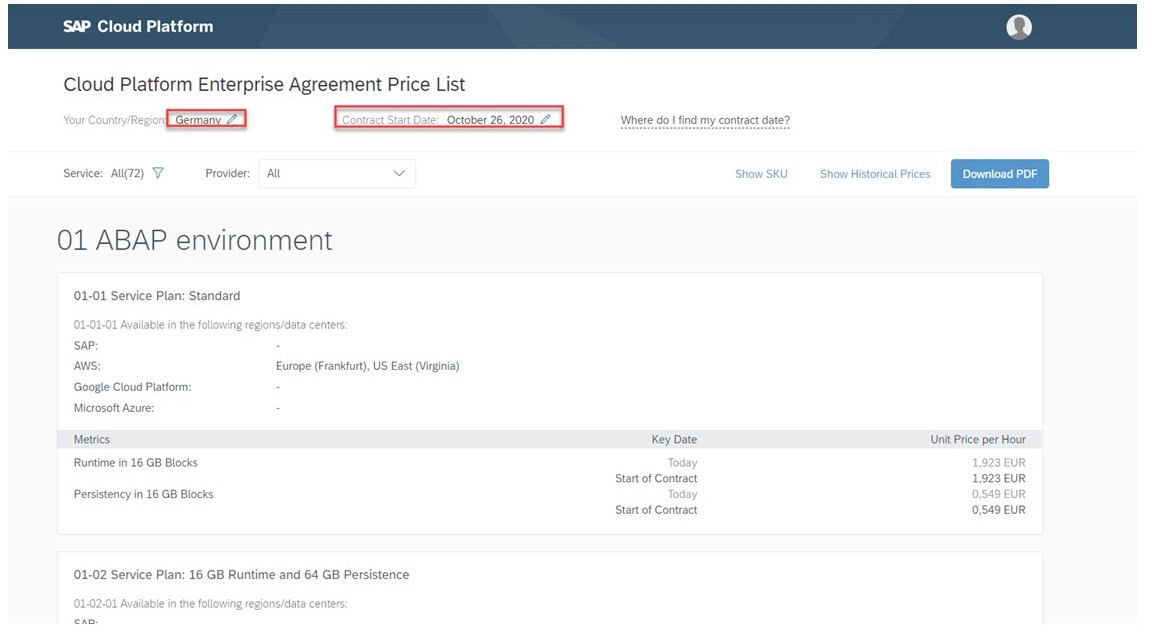
Under https://cloudplatform.sap.com/price-lists, you will find the currently valid price site, limited to countries.



Figure 36: An example for the ABAP service was in October 2020

In the upper area you can see the previously selected region and the date. A little deeper under the Show Historical Prices tab you can see the price changes and many more on this side.

Unit 3

Exercise 3

# Show Services at discover-center.cloud

## 67

1.

Unit 3

Solution 3

# Show Services at discover-center.cloud

## 68

1.

LESSON SUMMARY

You should now be able to:

* + Explain the Commercial Model of SAP Cloud Platform

Unit 3

Lesson 3

# Explaining the Runtime of Cloud Foundry

## 70

LESSON OBJECTIVES

After completing this lesson, you will be able to:

* Explain the account model in Cloud Foundry
* Explain the difference between platform and business users

### Cloud Foundry Account Model

In particular, we'll look at the following topics in this lesson:

* SAP Cloud Foundry in a Nutshell
* Cloud Foundry Account Model
* Global Account
* Subaccounts
* Orgs, and Spaces
* Directories (Beta) [Feature Set B]
* Custom Properties [Feature Set B]
* User Management
* Platform Users
* Business Users

SAP Cloud Foundry in a Nutshell SAP Cloud Foundry is:

* SAP Cloud Foundry should be used if you prefer a Managed Build-on approach. Cloud Foundry is fully managed by SAP and you benefit from a high level of abstraction from the underlying infrastructure.
* Cloud Foundry is the industry-standard open source cloud application platform for developing and deploying enterprise cloud applications.

You don't have to deal with technical aspects such as virtual machines, networking, monitoring, and more. It provides out-ofthe-box integration into all mandatory kernel services.

Cloud Foundry Account Model

In the following the account model.

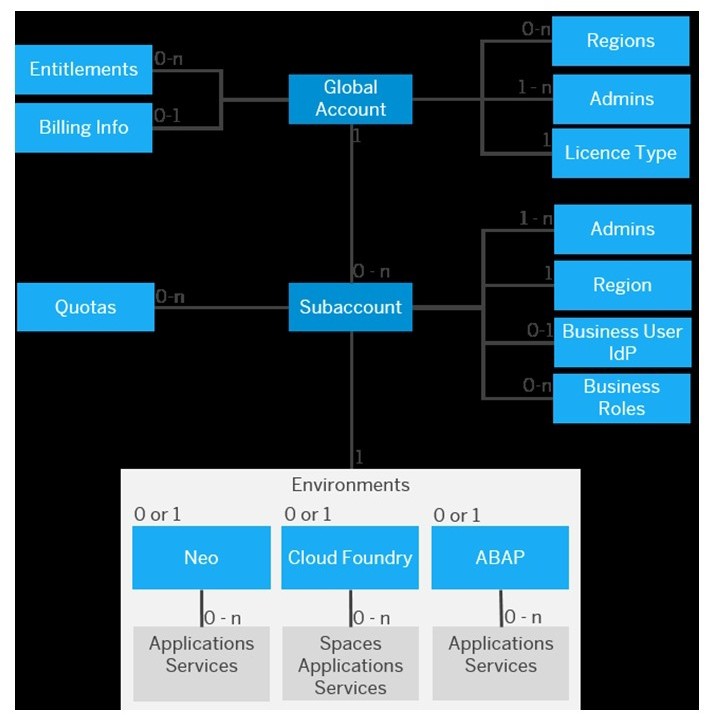


Figure 37: Account Model Overview

A global account has:

* + 0.n Entitlements
  + 0-n Regions ( based on the Subaccounts)
  + 1-n Members with the role admins On licence Type

A subaccount has:

* + 0-n Quotas
  + 1-n Business Users within the IDP
  + 0-n Business Roles
  + One Region

Cloud Foundry enviroment has:

* + 0-n Spaces
  + 0-n Applications
  + 0-n Services

Global Account

Facts about the Global Account:

It is the realization of a contract you made with SAP.

* + Global accounts are region- and environment-independent.
  + Within a global account, you manage all of your subaccounts, which in turn are specific to one region.



Figure 38: Global Account

A global account is used to manage subaccounts, members, entitlements and quotas. You receive entitlements and quotas to use platform resources per global account and then distribute the entitlements and quotas to the subaccount for actual consumption. There are two types of global accounts: enterprise accounts (paid) and trial accounts (free). The type determines pricing, conditions of use, resources, available services, scope of the functionality that you can use, and the level of support you can receive.

Sample

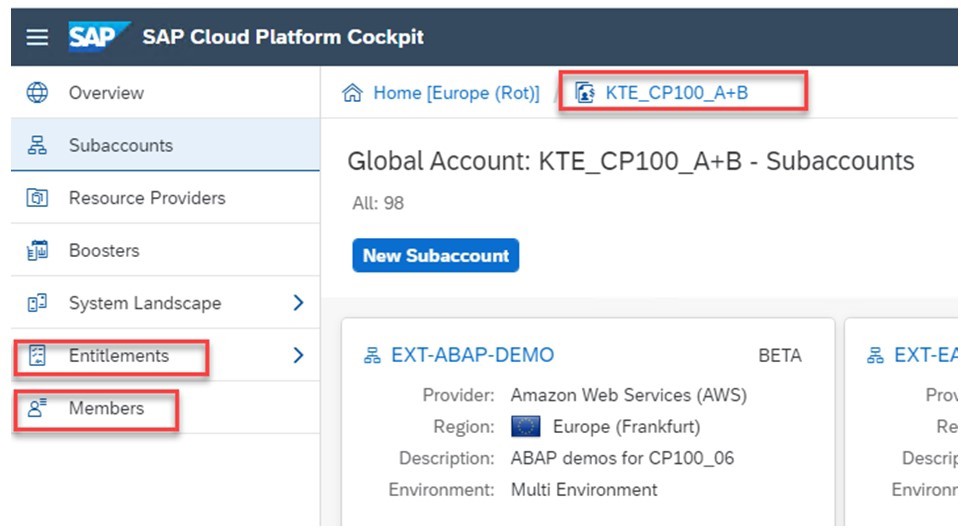
In the following a sample of an global account of an real enterprise cloud platform



Figure 39: A Global Account Named KTE\_CP100\_A+B With Entitlements and Members

Here you can see the cockpit of a global account. At the top you can see the breadcrumb navigation with the name of the global account. On the left side the navigation with the global members and entitlements on global level.

Subaccounts

Facts about Subaccounts:

* Subaccounts let you structure a global account according to your organization’s and project’s requirements with regard to members, authorizations, and entitlements.

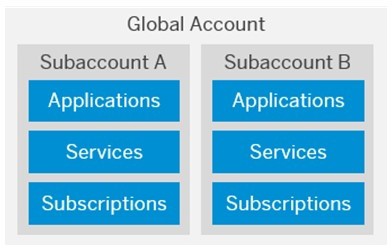


Figure 40: Sub Account

A global account can contain one or more subaccounts in which you deploy applications, use services, and manage your subscriptions. Subaccounts in a global account are independent from each other. This is important to consider with respect to security, member

management, data management, data migration, integration, and so on, when you plan your landscape and overall architecture.

Sample

In the following a sample of an sub account of an real enterprise cloud platform.

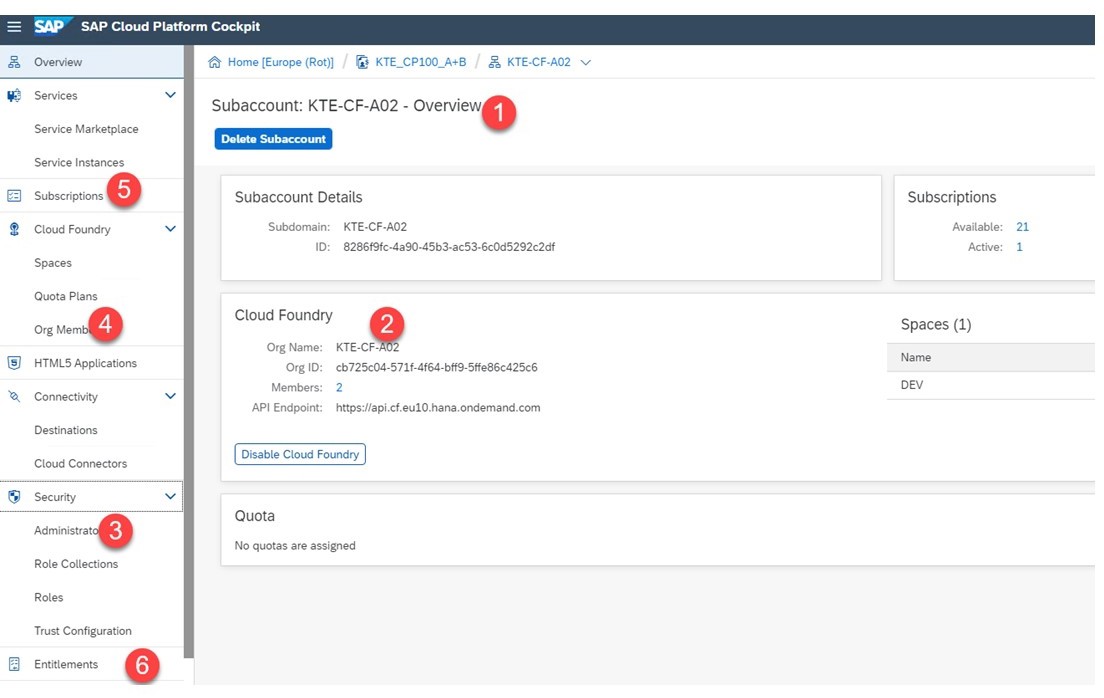


Figure 41: A Real Subaccount with Name KTE-CF-A02

Subaccount with:

1. Name
2. Runtime - here Cloud foundry
3. Security Admin environment independent
4. Org Member Cloud Foundry specifically
5. Apps as Subscriptions
6. Entitlements and Quotas

Orgs, and Spaces



Figure 42: Orgs and Spaces

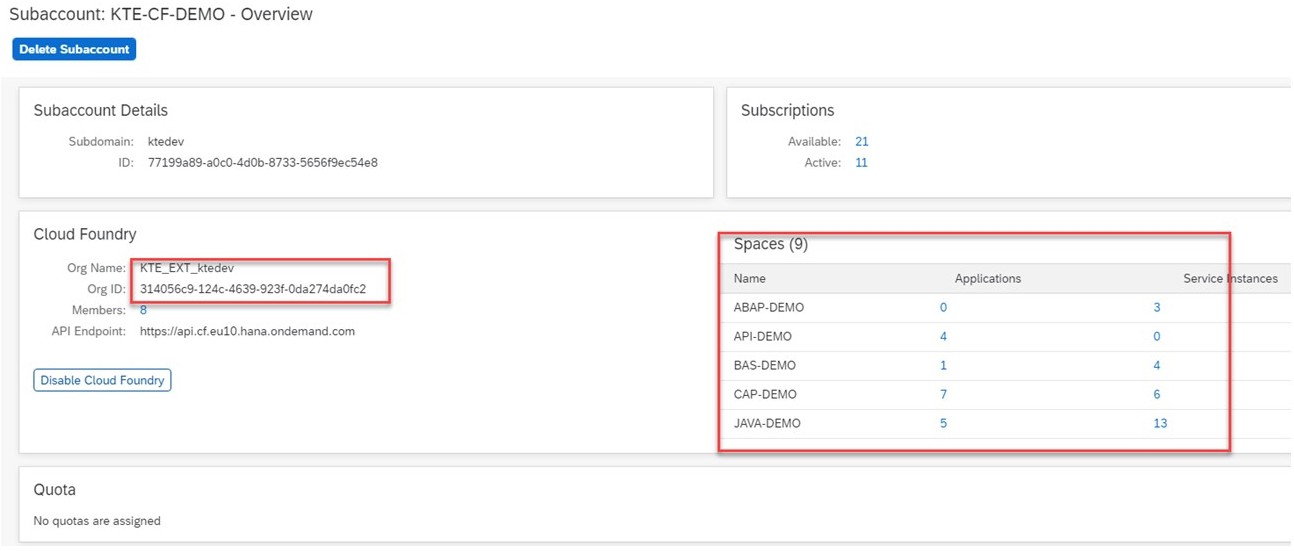
The subaccount and the org have a 1:1 relationship and the same navigation level in the cockpit (even though they may have different names). You can create spaces within that Cloud Foundry org. Spaces let you further break down your account model and use services and functions in the Cloud Foundry environment.

Sample

In the following a sample of an org and spaces of an real enterprise cloud platform is shown.



Figure 43: A Real Sample of Orgs and Spaces

Subaccount with:

* Org Name and ID
* With 9 spaces with deployed Apps

Directories (Beta) [Feature Set B]



Figure 44: Use of Directories

Directories allow you to organize and manage your subaccounts according to your technical and business needs.

A directory can contain one or more subaccounts. It cannot contain other directories. Using directories to group subaccounts is optional - you can still create subaccounts directly under your global account.

Custom Properties [Feature Set B] Use of Custom Properties:

Table 1: Use of Custom Properties

|  |  |
| --- | --- |
| Custom Property (Name) | Property Values |
| Landscape | Dev, Test, Production |
| Department | Hr, IT, Finance, Sales |
| Cost Center | 000001134789, 000002155534, To be de-  fined |
| Flagged for Deletion | (no values) |
| Important | (no values) |

Custom properties allow you to label or tag your directories and subaccounts according to your own business and technical needs. This makes organizing and filtering your directories and subaccounts easier within your global account.

You create and assign custom properties when you create or edit a directory or subaccount. Using custom properties is optional.

Unit 3

Exercise 4

# Show aSubaccount with Cloud Foundry

## 77

1.

Unit 3

Solution 4

# Show aSubaccount with Cloud Foundry

## 78

1.

### Users in Cloud Foundry

A user account corresponds to a particular user in an identity provider, such as the SAP ID service and consists, for example, of an SAP user ID (S-user or P-user) and password.

User accounts enable users to log on to SAP Cloud Platform and access subaccounts and use services according to the permissions given to them.

It's important to understand the difference between the 2 types of users we are referring to:

* + platform users
  + and business users.

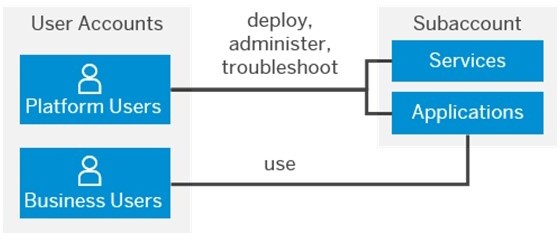


Figure 45: Different Types of Users

Platform users deploy , adminster and create apps and uses services to integrate or extend business functionality. Business User use this created apps- Both need to authenticate on a central place, for example SAP Identity Provider.

Platform Users

Facts about Users in Cloud Foundry:

* + Platform users are usually developers, administrators or operators who deploy, administer, and troubleshoot applications and services on SAP Cloud Platform.

They're the users that you give certain permissions for instance at global account or subaccount level, either by adding them as members with certain permissions or by assigning role collections to them .

Platform users who were added as members and who have administrative permissions can view and/or manage the list of global accounts, subaccounts, and Cloud Foundry orgs and spaces that are available to them, and access them using the cockpit or the command line interface.

For platform users, the default identity provider is SAP ID Service, but if you want to have subaccount members from your own user base, you can use your own identity authentication tenant with SAP Cloud Identity Authentication Service.

Sample - Authentication of platform users

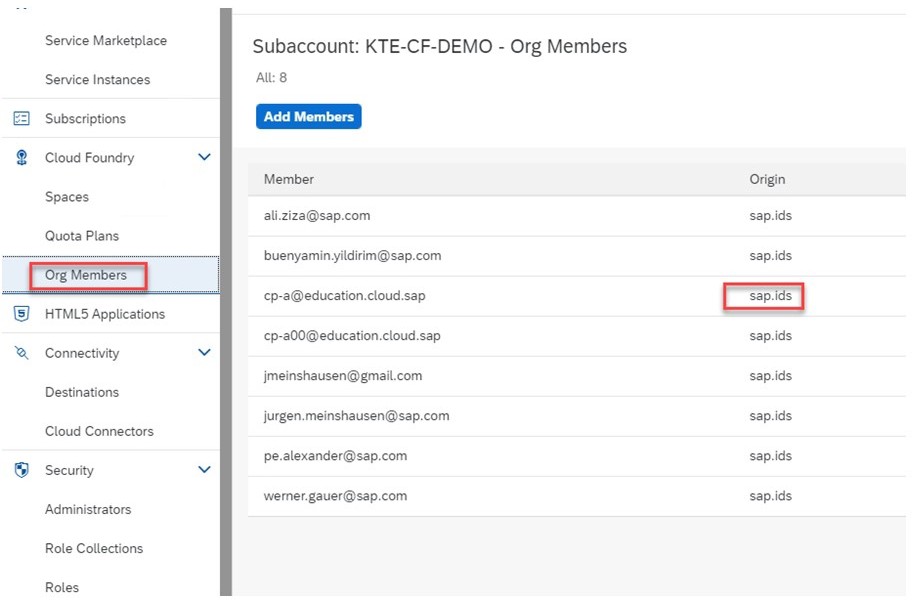


Figure 46: A Real Subaccount Which has Users Authenticated by the SAP ID Service

Org Members - Platform Users authenticated within the SAP ID Service.

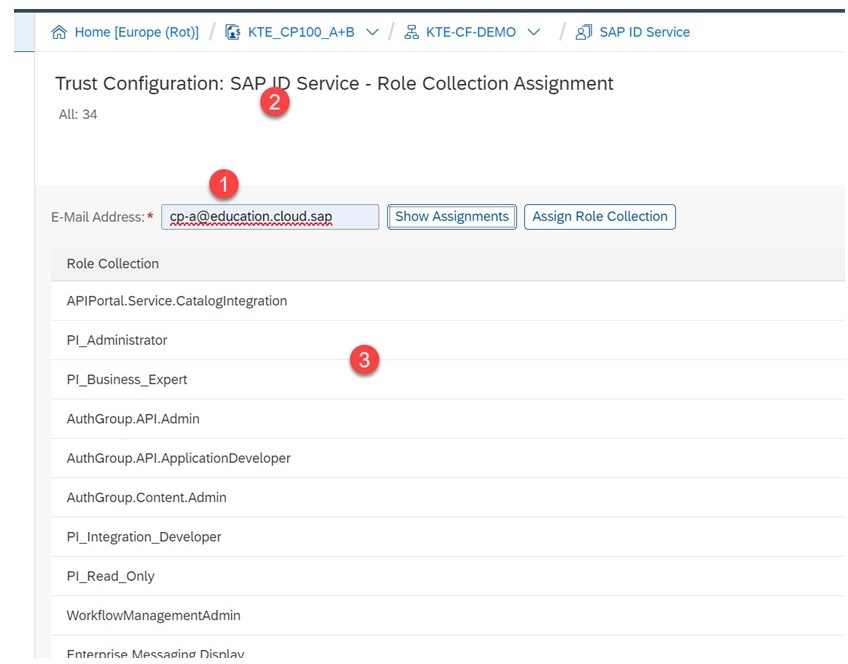
Sample - Autorisation of platform users



Figure 47: Authorisation of a Platform User With Roles Coming From Applications

Configured Platform users with role collections coming from applications. Explanations about the numbers:

1. The platform-user name.
2. The platform-user has to authenticated by the SAP ID Service.
3. The roles collections coming from the SaaS Applications like BAS, Integration Suite , IOT and others.

Business Users

Business users use the applications that are deployed to SAP Cloud Platform:

* + For example, the end users of your deployed application or users of subscribed apps or services, such as SAP Business Application Studio or SAP Web IDE, are business users.

In the Cloud Foundry environment, application developers (platform users) create and deploy application-based security artifacts for business users. Administrators use these artifacts to assign roles, build role collections, and assign these role collections to business users or user groups. In this way, they control the users' permissions in the deployed application.

For business users, the identity provider can be, for example, SAP Cloud Identity Authentication Service or your own, such as Active Directory.

LESSON SUMMARY

You should now be able to:

* Explain the account model in Cloud Foundry
* Explain the difference between platform and business users

Unit 3

Lesson 4

# Exploring the Service: ABAP

## 83

LESSON OBJECTIVES

After completing this lesson, you will be able to:

* + Explore the most important components of the ABAP installation

### ABAP Service

In particular, we'll look at the following topics in this lesson:

* + Introduction
  + The main building blocks of ABAP in the Cloud.
  + How are the look and feel.
  + Step by step Sample.

Introduction ABAP in the cloud:

* + Is a Platform as a Service (PaaS) offering for ABAP.
  + Develop ABAP cloud apps decoupled from the digital core Leverage your ABAP know how in the cloud Reuse your existing ABAP assets.
  + Benefit from newest ABAP Programming Model Exploit SAP HANA capabilities Consume SAP Cloud Platform services.

Facts about ABAP in the cloud

The ABAP Platform provides the technology layer for several SaaS applications like S/4HANA Cloud and Integrated Business Planning as well as for on-premise solutions like S/4HANA.

Furthermore, the ABAP Platform is also used as the key pillar of SAP's Platform as a Service offering for ABAP which has the product name SAP Cloud Platform, ABAP environment.

Using the same code line for these different flavors enables customers and partners to use the same powerful toolset (ABAP Development Tools in Eclipse) and the same ABAP RESTful Programming Model (RAP) for cloud and on-premise development.

There are two main reasons to develop applications and extensions for the cloud with ABAP: Existing assets (custom code developed in on-premise systems) and developers, experienced in using ABAP. The typical extension scenarios built on top of ABAP are:

* + Cloud ERP: Extend SAP S/4HANA Cloud or other SAP cloud offerings with cloud extensions.
* Innovation Platform: Develop and run innovative ABAP apps on a PaaS ( SAP Cloud Platform) in the Cloud.
* Hub-like Usage: Integrate multiple cloud & on-premise systems with SAP & non-SAP cloud services.

The ABAP service runs on the Cloud Foundry Environment and must be configured there on the desired subaccount.

The main building blocks of ABAP in the Cloud

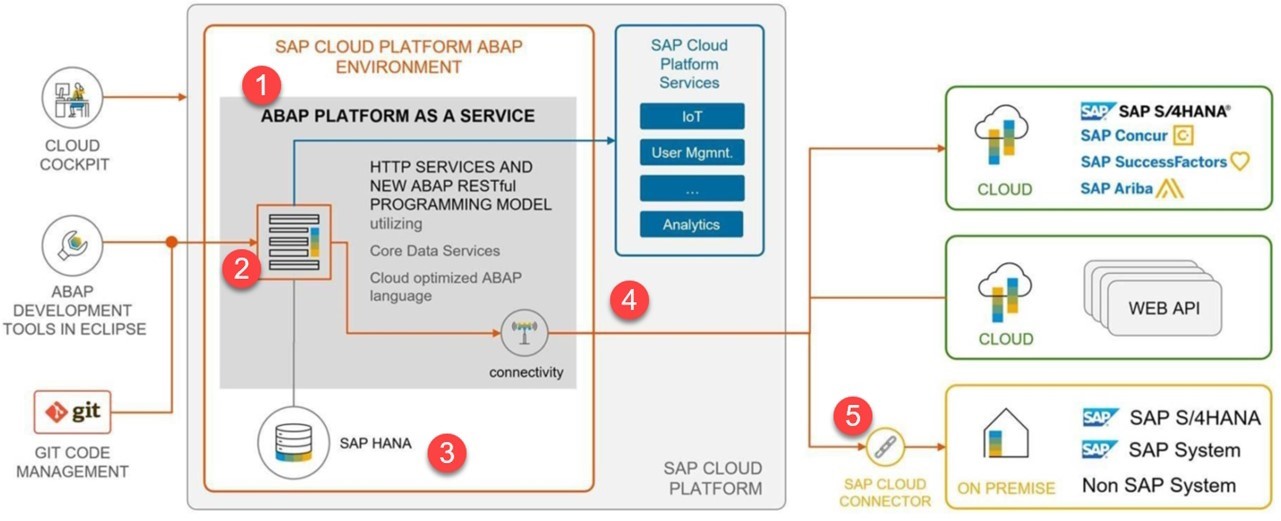
Below is the overview of the main building blocks of ABAP in the Cloud.



Figure 48: The Main Building Blocks of ABAP in the Cloud

Explanations:

1. An ABAP instance runs on SAP CP Cloud Foundry as a Service.
2. There is a SaaS Application which connect to the Development tools.
3. You use only a HANA Database with all the advantages of HANA.
4. You can use all available services on the SAP CP like connectivity or IoT.
5. You can use the cloud connector to connect to back ends.

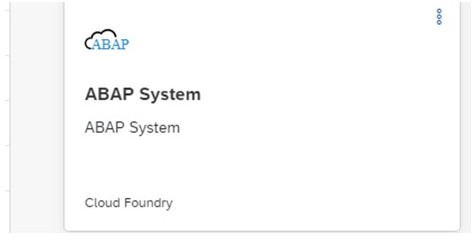
How are the look and feel

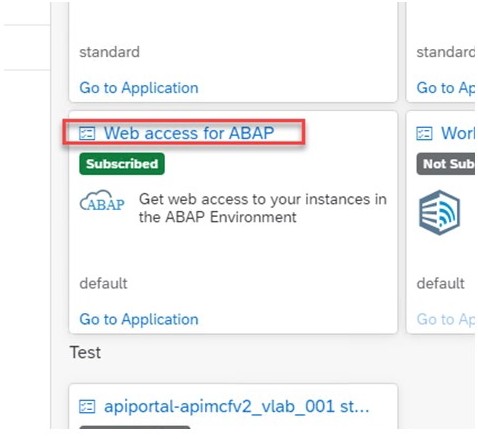


Figure 49: Service ABAP System in Cloud Foundry

ABAP service as a tile in the SAP Cloud Platform.



Figure 50: SaaS App Web Access for ABAP in Cloud Foundry

You will see the Web access for ABAP at *Subscriptions*.

Step by step sample



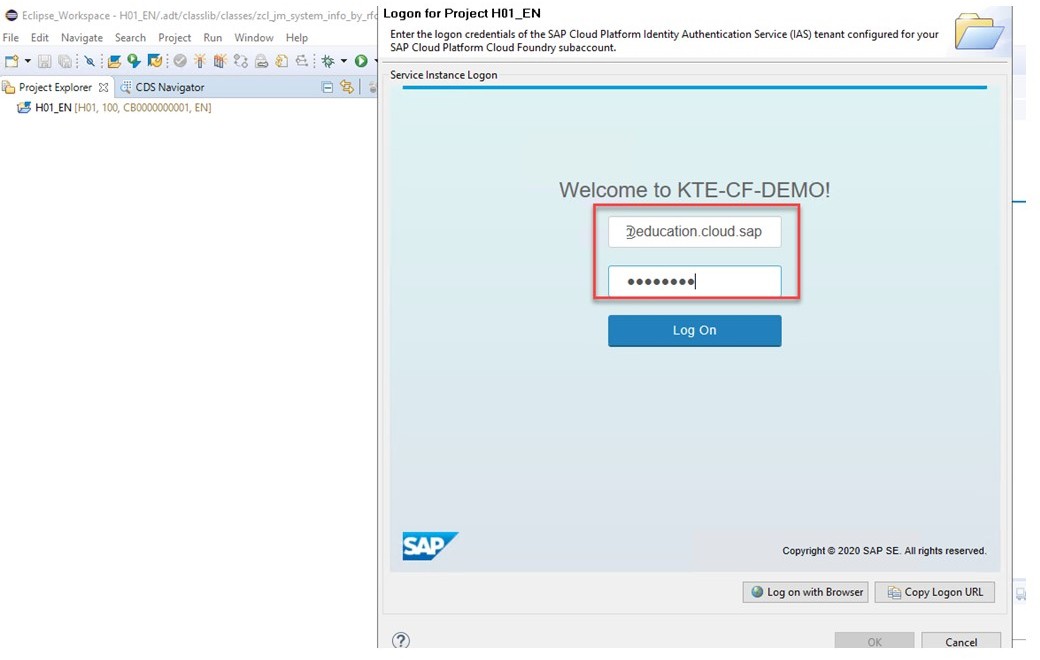
Figure 51: ABAP Service

Explanations:

* 1. Create and configure your Cloud Foundry Subaccount on SAP Cloud Platform.
     + Assign the ABAP Service.
     + Assign the SaaS app Web access for ABAP.
  2. Instance your ABAP Service with the instance wizzard
  3. Download and open an Eclipse for ABAP.



Figure 52: ABAP Service

* 1. Connect against your configured subaccount with the ABAP service.

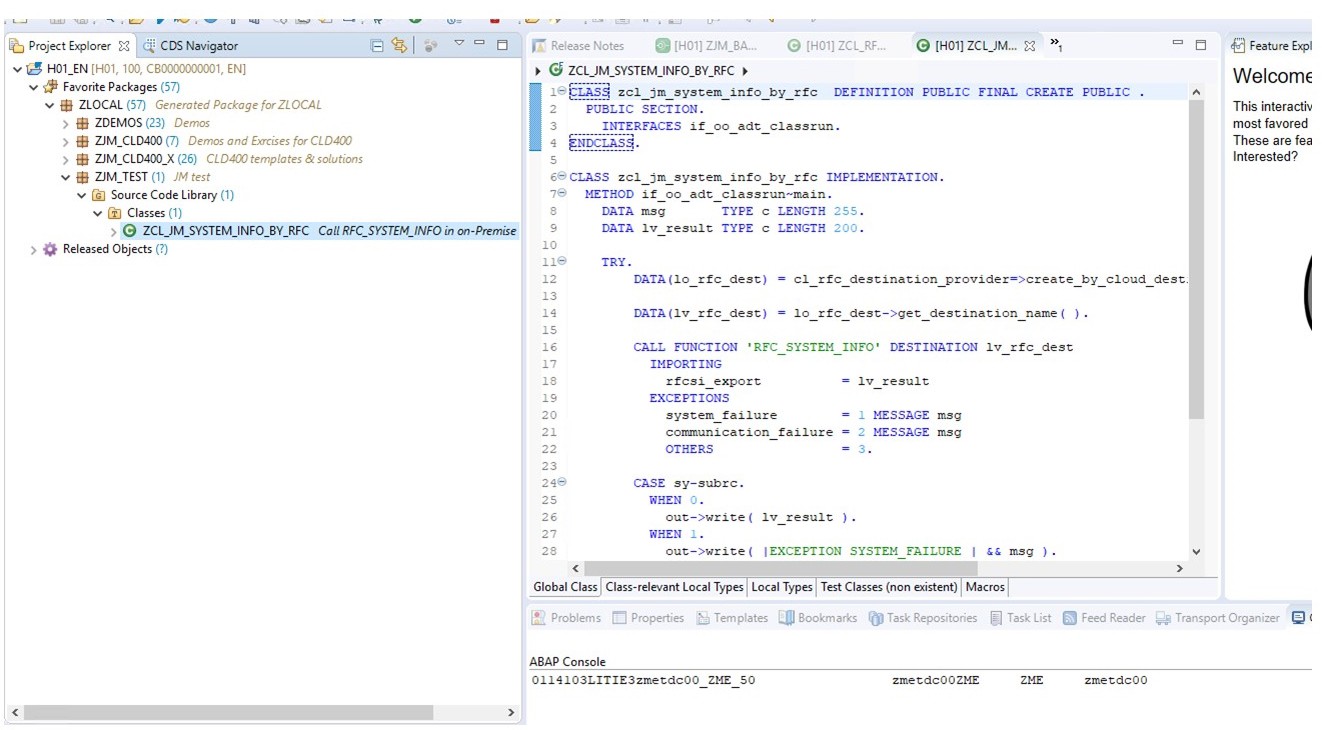
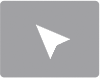


Figure 53: Connecting Against the ABAP Service

* 1. Code and run your app.

How to Use ABAP for SAP Cloud Platform What are you going to show?

* Run an ABAP class that provides information from an R3 RFC connected via cloud

connector.

* The ABAP Stack is located on the Cloud Foundry - ABAP in the Cloud.
* Complete Architecture.

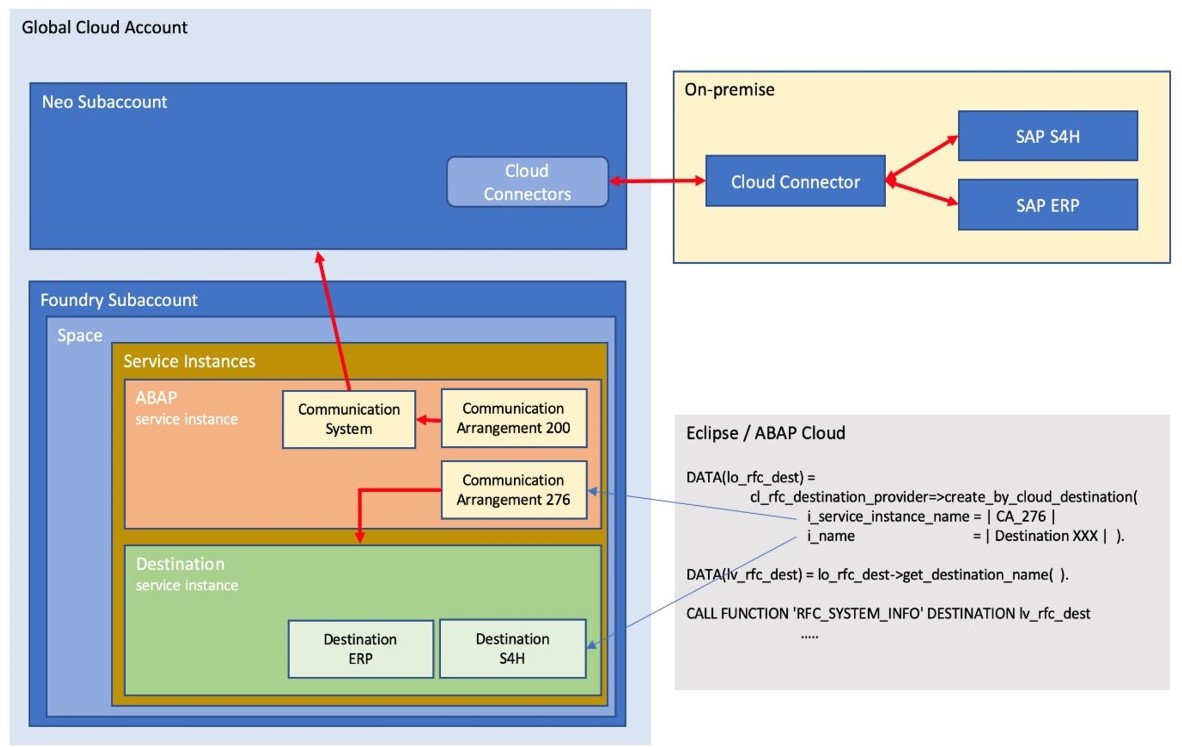


Figure I-2: Architecture of the Demonstration

This Demonstration consists of several parts:

* + Run the scenario: Step 1
  + Show the implementation: Step 2
  + Show the RFC Call at the ZME- back end: Step 3
  + Show the cloud connector: Step4
  + Show the destination at KTE-NEO-DEMO: Step 5
  + Show the ABAP instance at KTE-CF-DEMO: Step 6
  + Further explanations: Step 7



Note:

In the exercise Assign To and Set up Your Training Environment work folders are created. In these work folders, you find all required information to successfully perform this demonstration. Please perform this exercise first or refer to the SSG.



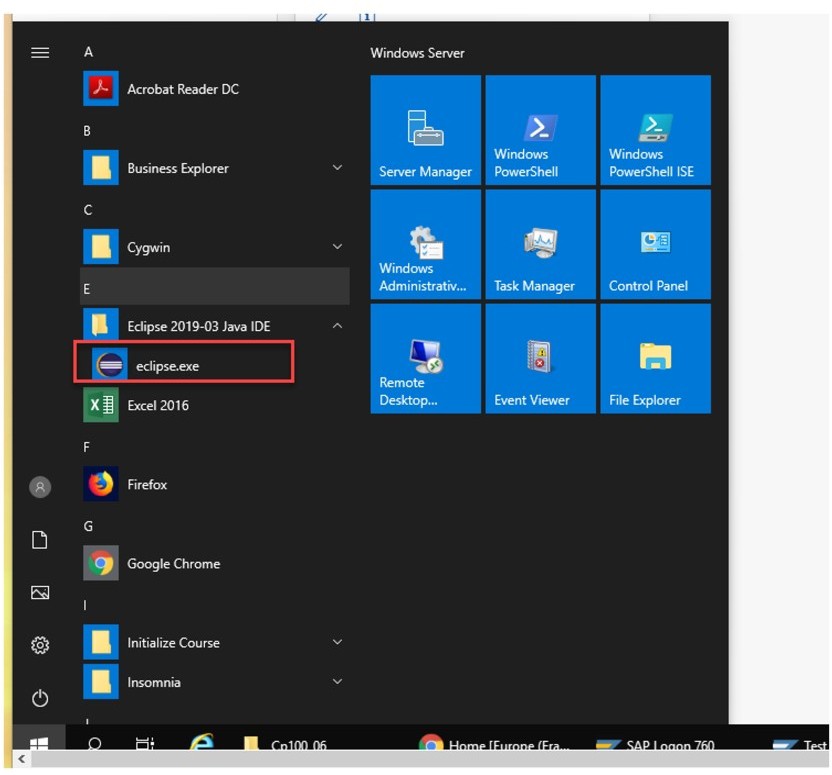
Note:

The following demo is performed in the training landscape dy-ecc617ciscc-###.

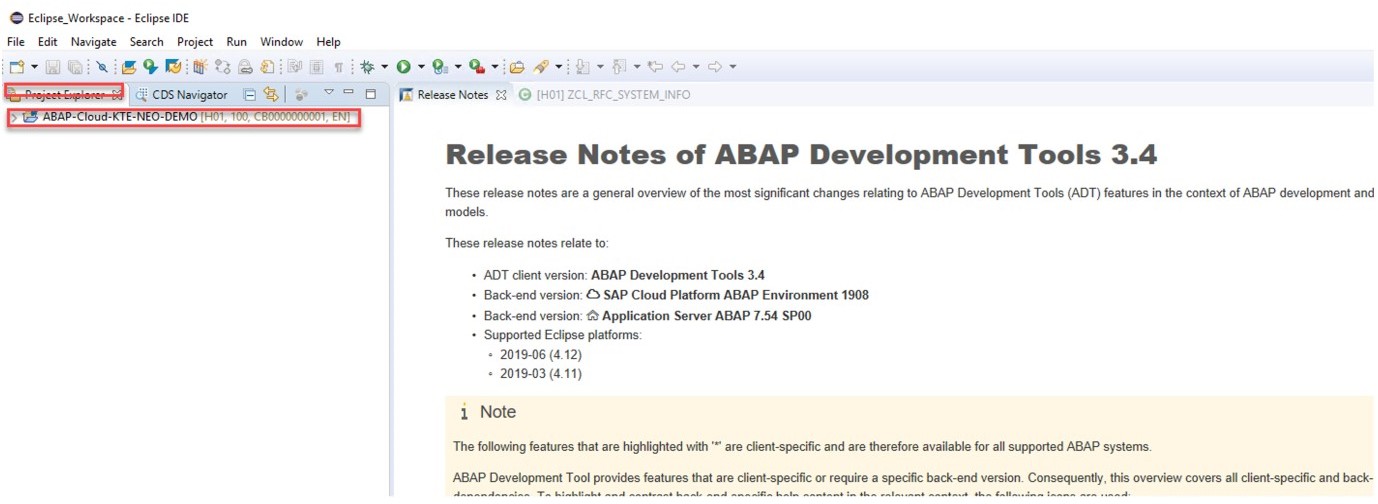
1. a) Log in to *dy-ecc617ciscc-clouddemo*.

You find the required credentials in the SSG.

* 1. Click on the *Eclipse.exe* tile to start *eclipse die 2019-03 Java*.



* 1. Choose the *ABAP Perspective* if not predefined.
  2. On the left side, choose *Project Explorer*.



* 1. Click on your ABAP Project *ABAP-CLOUD.KTE-NEO-DEMO*.
  2. Use the following data:

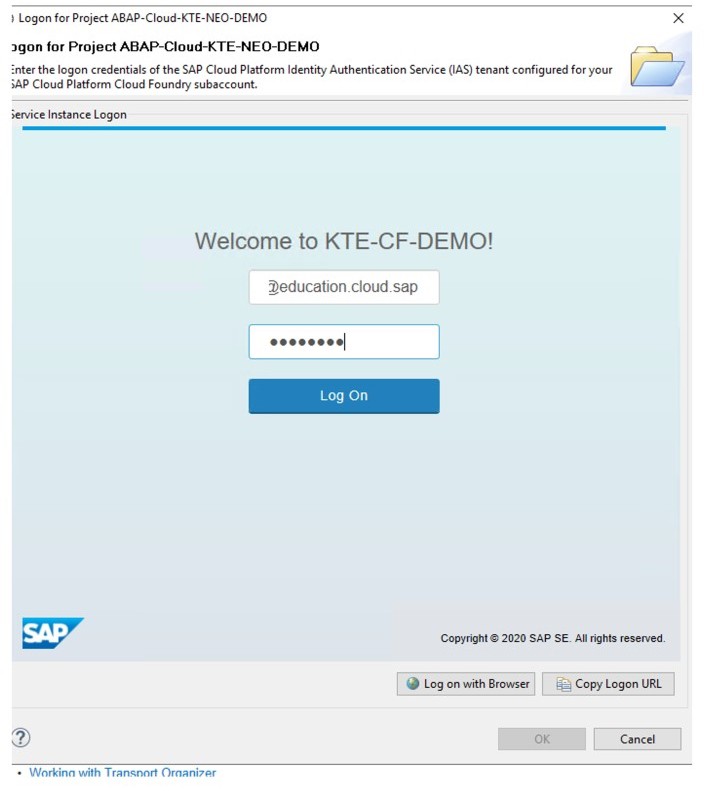
|  |  |
| --- | --- |
| Field | Value |
| User | [cp-a@education.cloud.sap](mailto:cp-a@education.cloud.sap) |
| Password | Welcome1 |



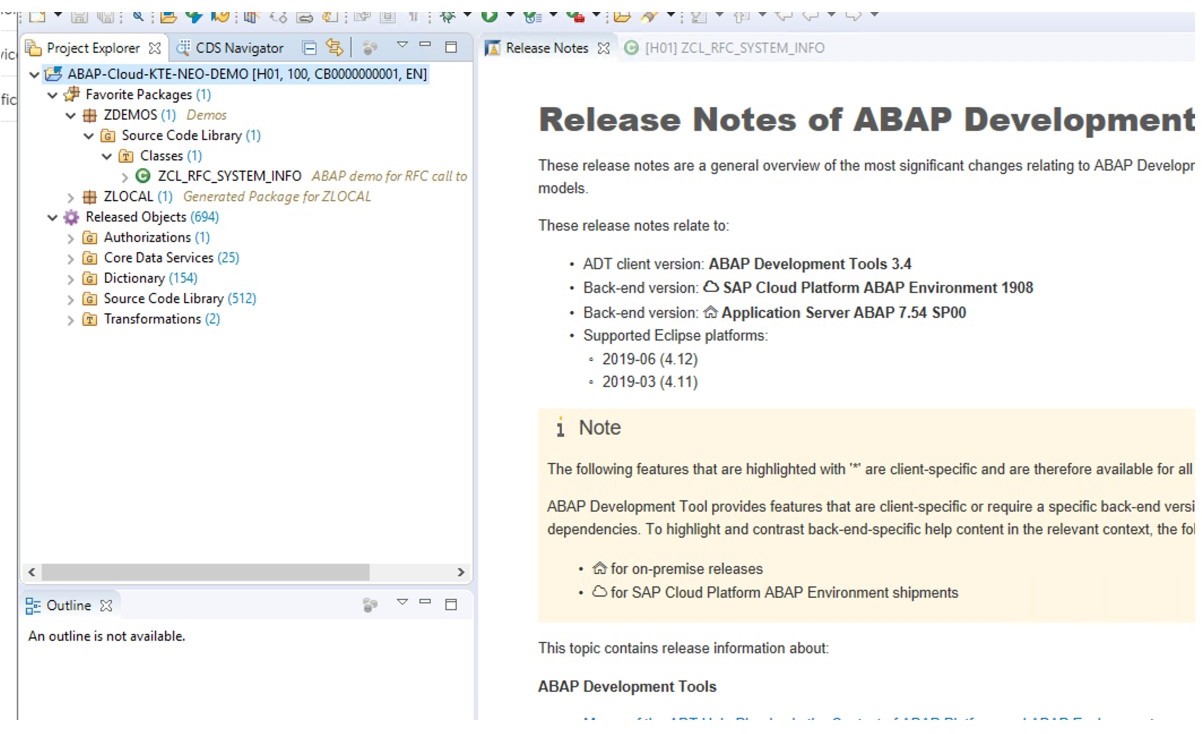
Note:

This information is subject to change. in case of issues, refer to the SSG.

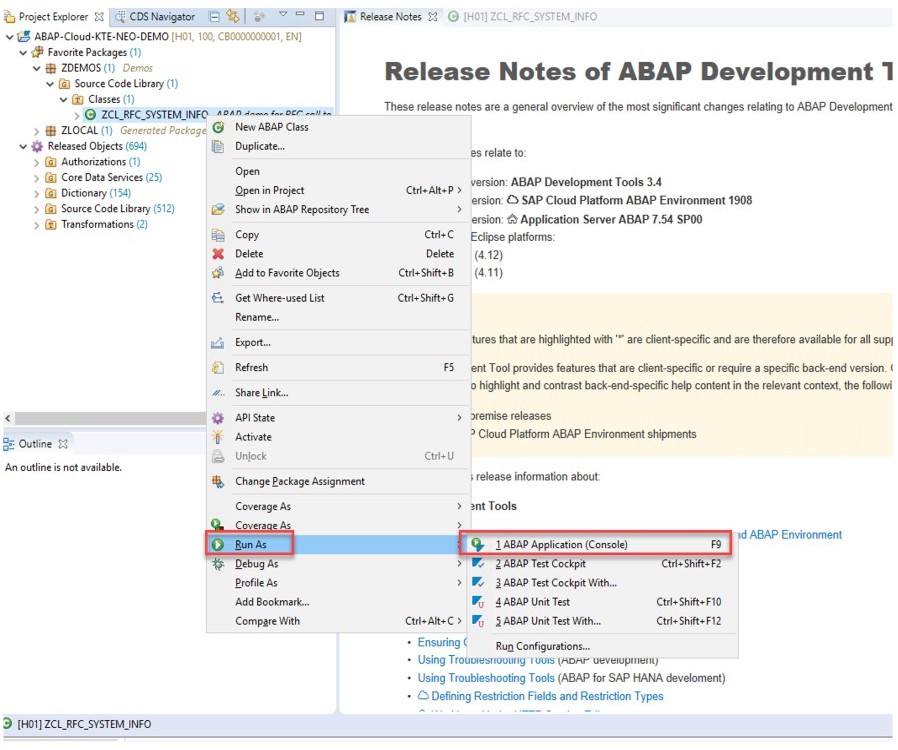
* 1. Click on *Log On*.



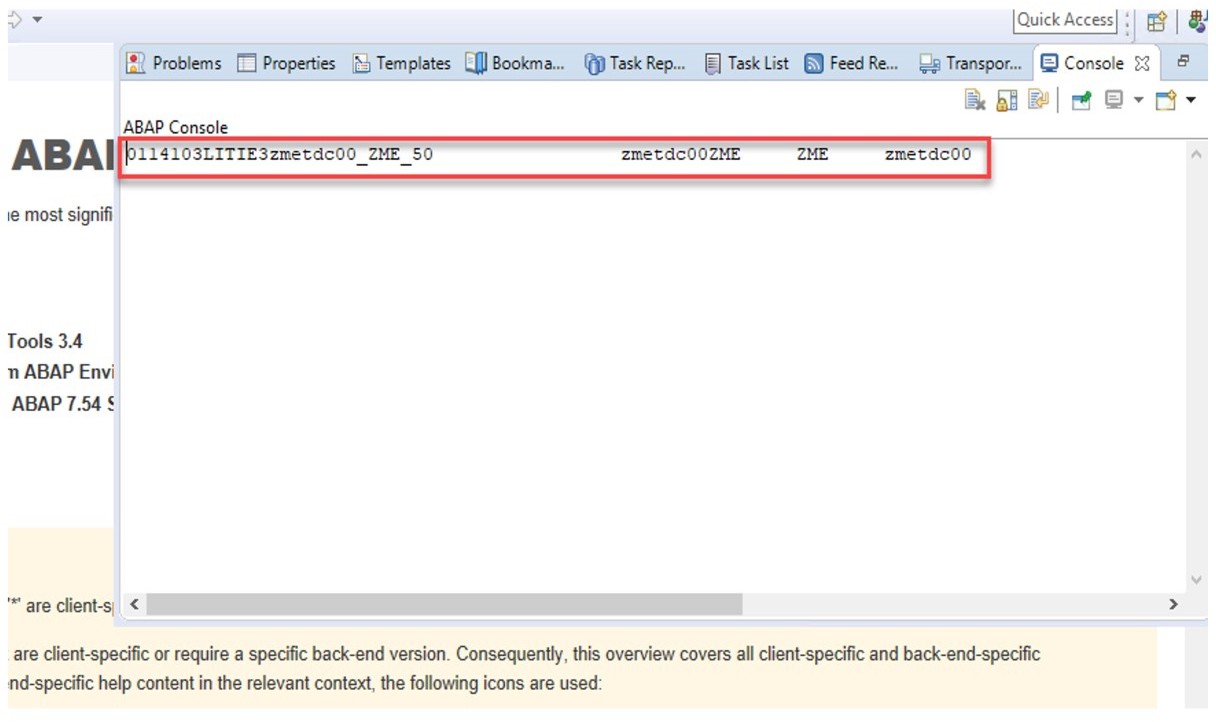
* 1. After that you have a running Cloud ABAP Instance hosted at *KTE-EXT* → *EXT-CF- DEMO* .
  2. Navigate to the class *ZCL\_RFC\_SYSTEM\_INFO*.



* 1. Choose *F9* to test the class run.

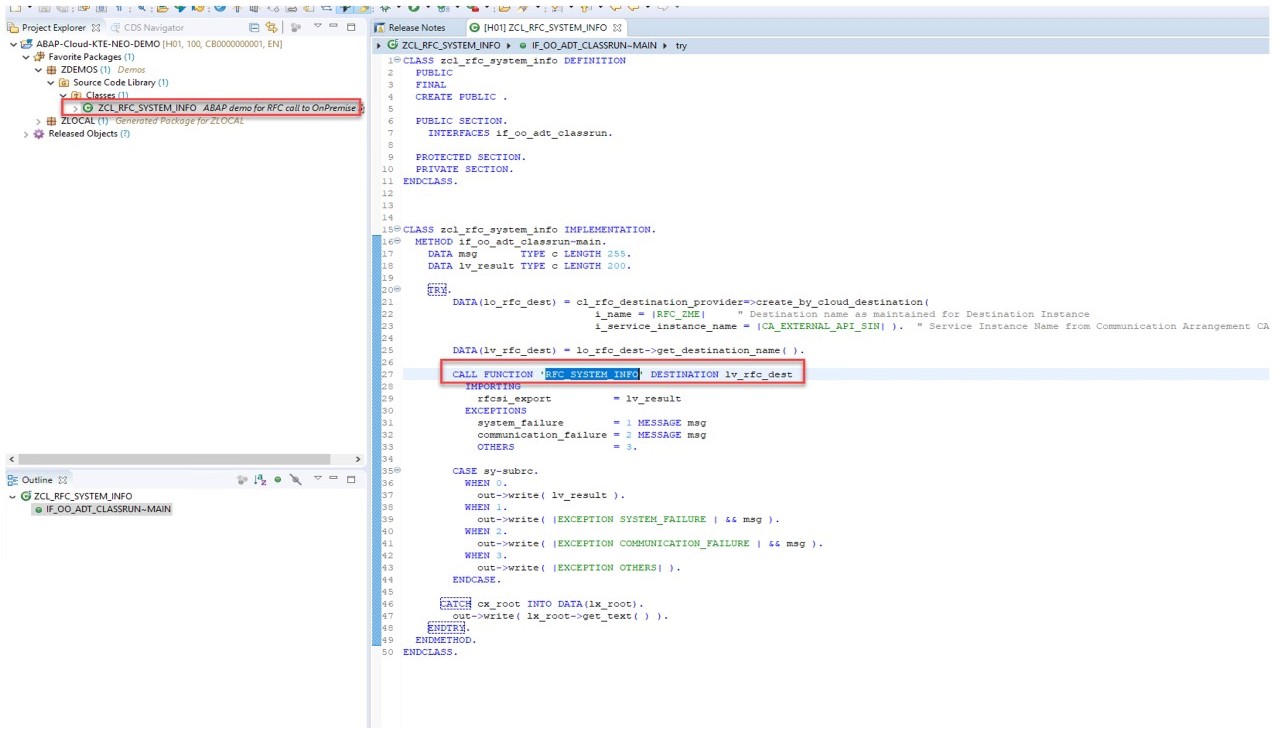


* 1. The result of the RFC call from ZME back end displays in the console:

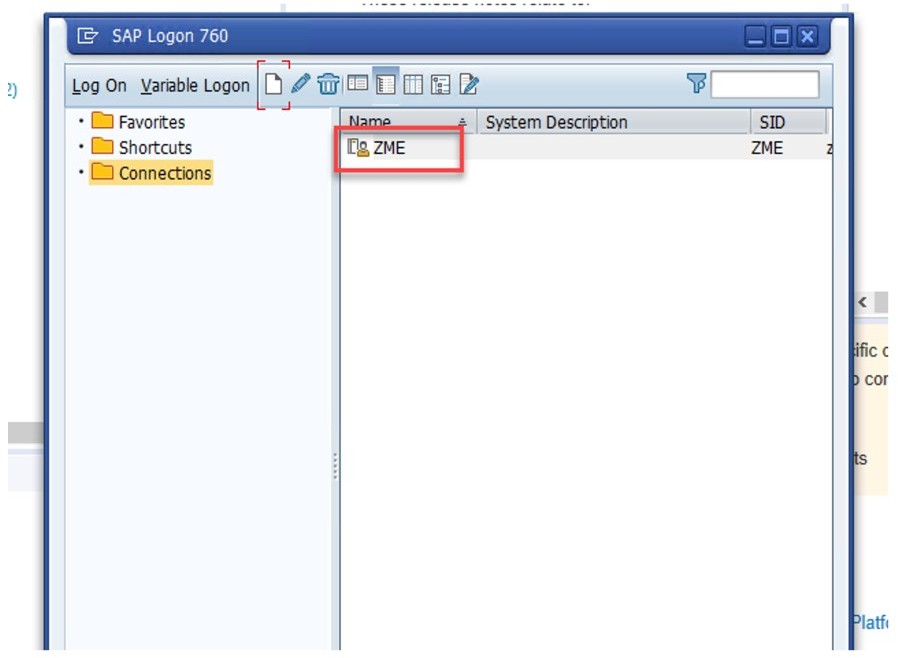


1. Show the Implementation.

a) Open the class *ZCL\_RFC\_SYSTEM\_INFO* and show the RFC call.



1. Show the RFC call at the ZME- back end.
2. You are in *dy-ecc617ciscc-clouddemo*.
3. Open the *SAP Logon* and choose the system *ZME*.



1. Log in with the following credentials:

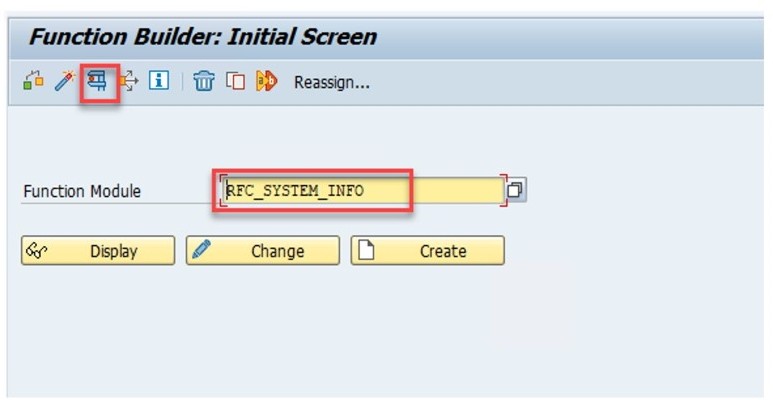
|  |  |
| --- | --- |
| Field | Value |
| User | cldtrain-00 |
| Password | Welcome1 |



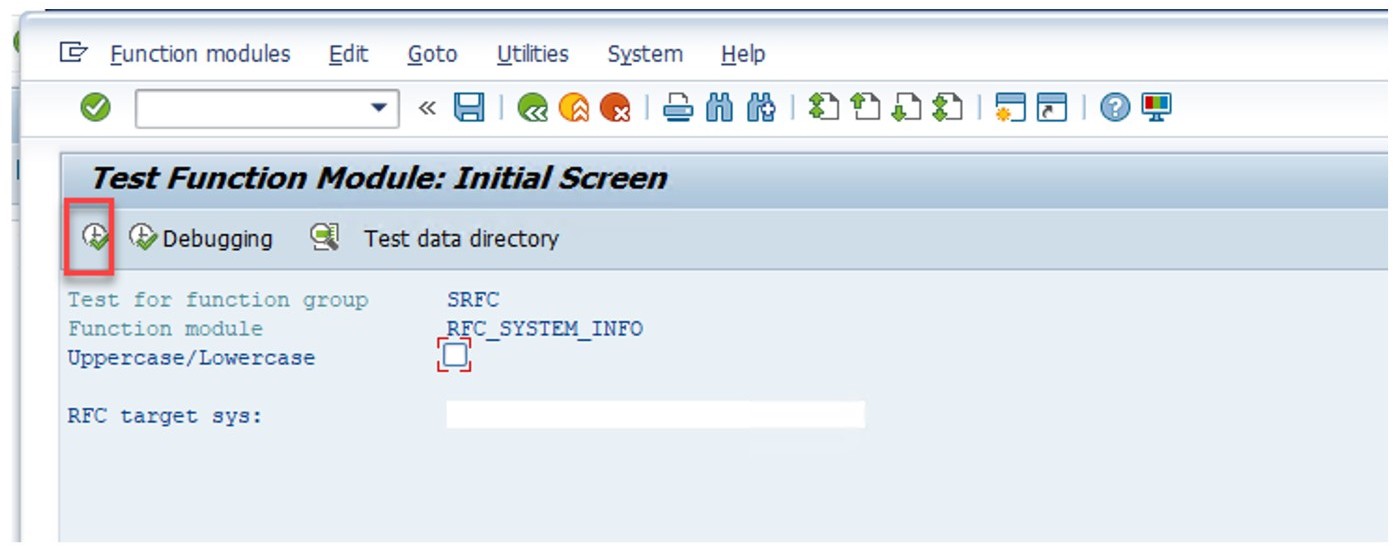
Note:

This information is subject to change. in case of issues, refer to the SSG.

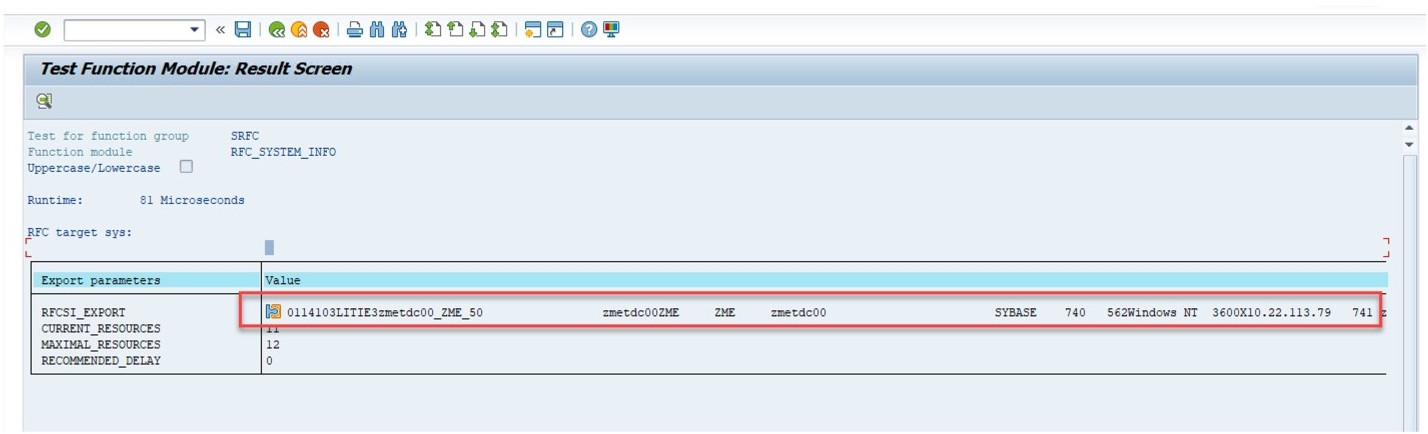
1. Execute transaction *SE37*.
2. In the *Function Module* field, enter **rfc\_system\_info**.
3. Click on *Test/Execute*.



1. On the next screen, click on *Test/Execute*.



1. Now find the same results as before:



1. Show the Cloud Connector.
2. You are in *dy-ecc617ciscc-clouddemo*.
3. Open the Chrome Browser and enter https://wdflbmt7269:8443
4. Logon with the following credentials:

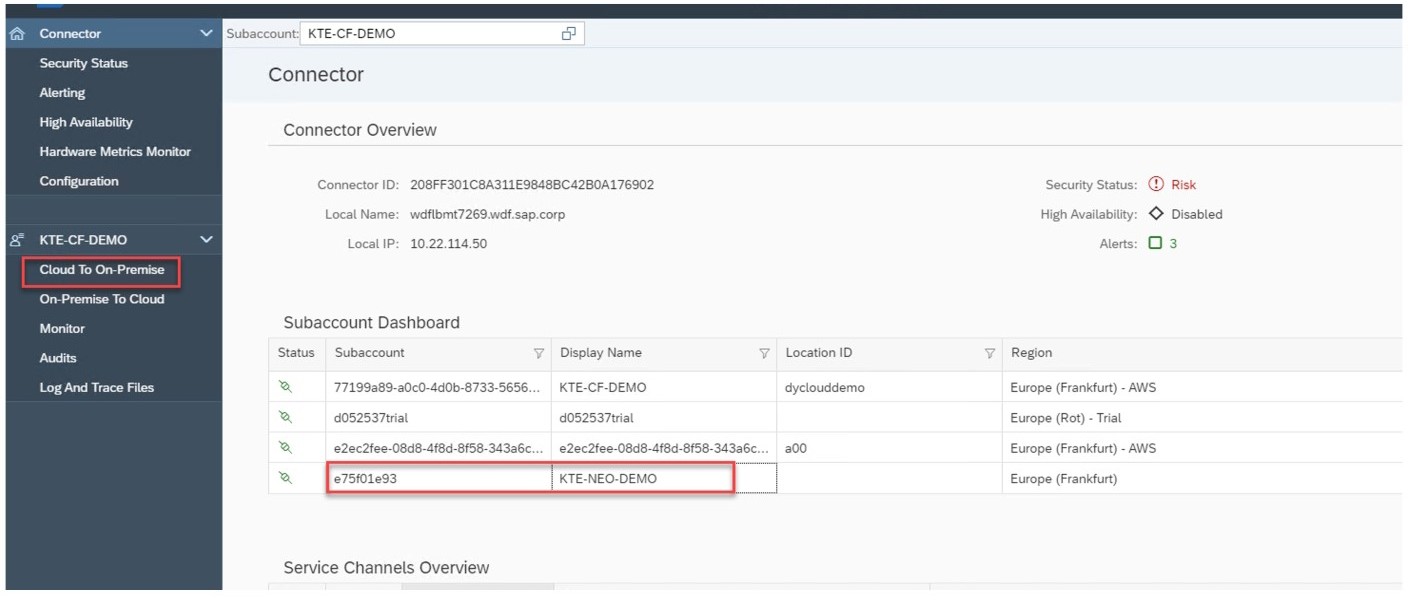
|  |  |
| --- | --- |
| Field | Value |
| User | Administrator |
| Password | sap123 |



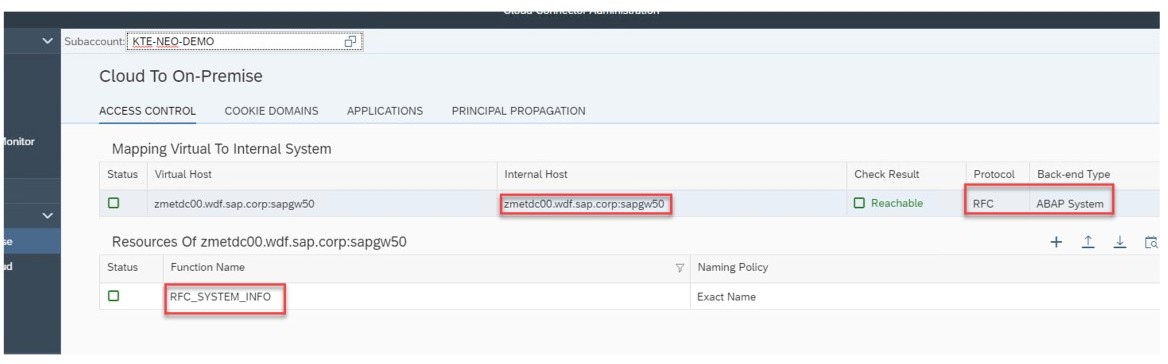
Note:

This information is subject to change. in case of issues, refer to the SSG.

1. Find *KTE-NEO-DEMO* , mark it and click on *Cloud TO On-Premise*.



1. See the configured path to the ZME back end:



1. Show the Destination at KTE-NEO-DEMO.
2. Navigate to the NEO subaccount KTE-NEO-DEM0O: https:// account.eu2.hana.ondemand.com/cockpit#/globalaccount/822af597-cc75-4fbe- b83d-71c32cf0394a/neosubaccount/3b7fc05c-abb4-4ebe-82ab-9e3fa62566da.
3. As credentials use:

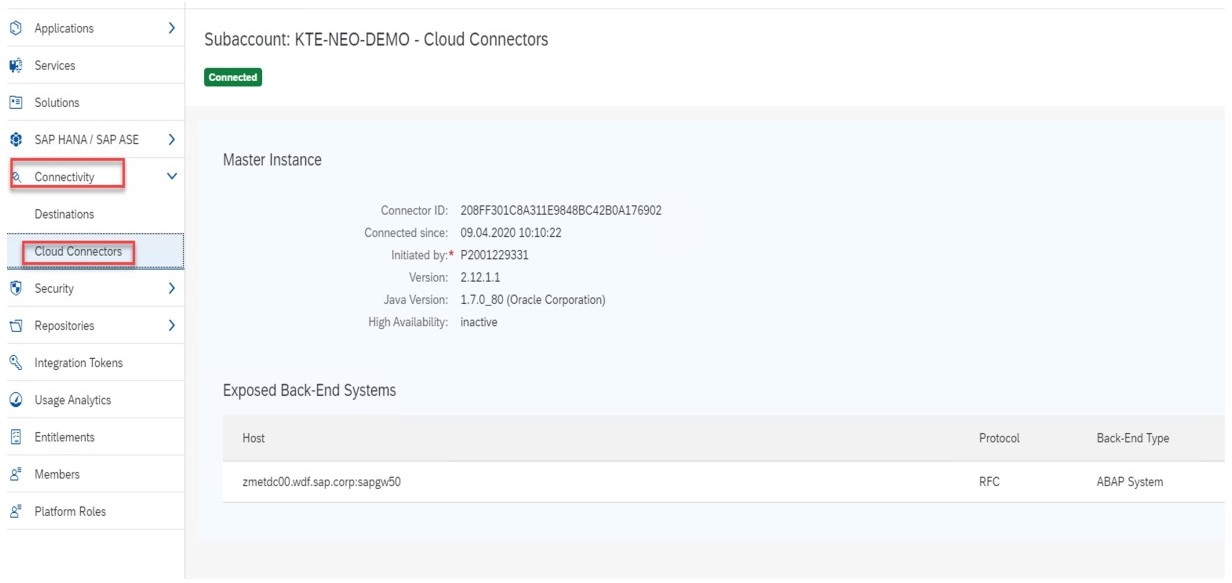
|  |  |
| --- | --- |
| Field | Value |
| User | [cp-a@education.cloud.sap](mailto:cp-a@education.cloud.sap) |
| Password | Welcome |



Note:

This information is subject to change. in case of issues, refer to the SSG.

1. Open the *Connectivity* → *Cloud Connectors* and show the Cloud Connector Connection.



1. Show the ABAP Instance at KTE-CF-DEMO.
2. Navigate to *KTE-CF-DEMO* → *ABAP-DEMO*: https:// account.eu2.hana.ondemand.com/cockpit#/globalaccount/822af597-cc75-4fbe- b83d-71c32cf0394a/subaccount/77199a89-a0c0-4d0b-8733-5656f9ec54e8/org/ 314056c9-124c-4639-923f-0da274da0fc2/space/f1d135ce-6770-40dd-8ed4- bda7a949b83b/serviceinstances.
3. Logon with:

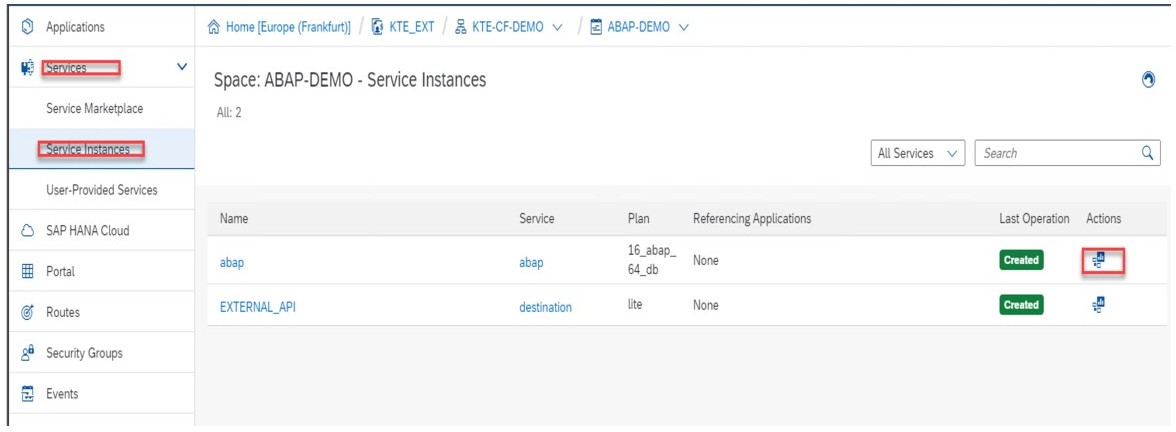
|  |  |
| --- | --- |
| Field | Value |
| User | [cp-a00@education.cloud.sap](mailto:cp-a00@education.cloud.sap) |
| Password | Welcome 1 |



Note:

This information is subject to change. in case of issues, refer to the SSG.

1. Navigate to *Services* → *Service Instances* and find the abap Instance.



1. Click on the Icon on the right side.
2. Log on with:

|  |  |
| --- | --- |
| Field | Value |
| User | [cp-a@education.cloud.sap](mailto:cp-a@education.cloud.sap) |

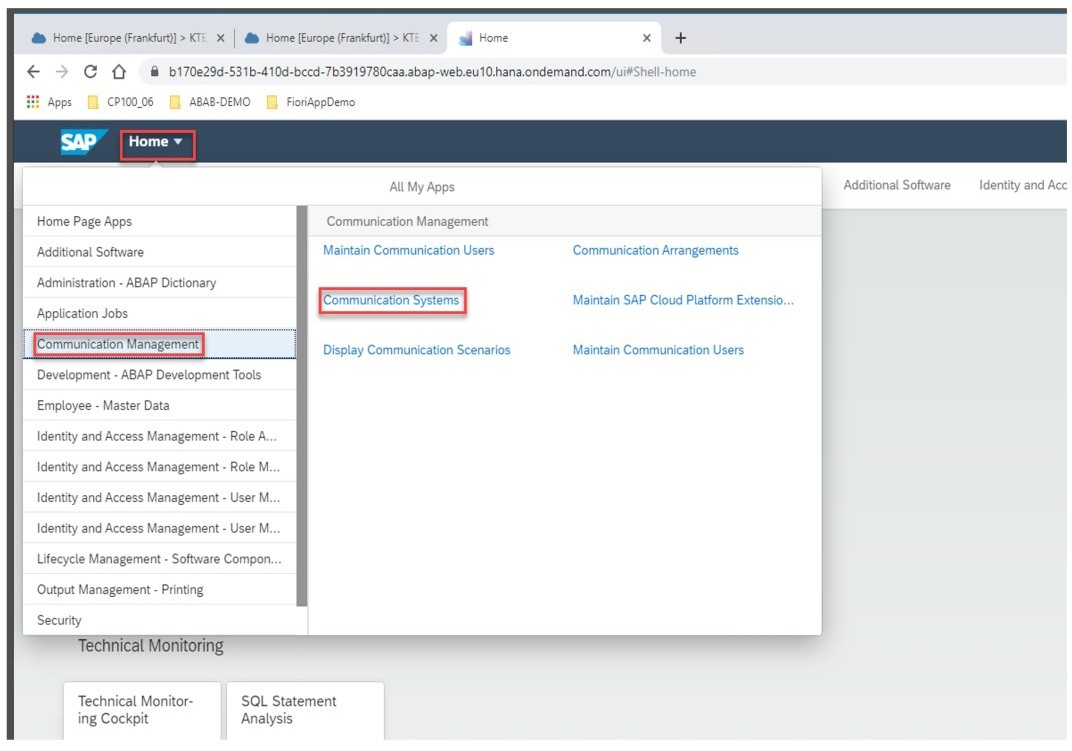
|  |  |
| --- | --- |
| Field | Value |
| Password | Welcome 1 |



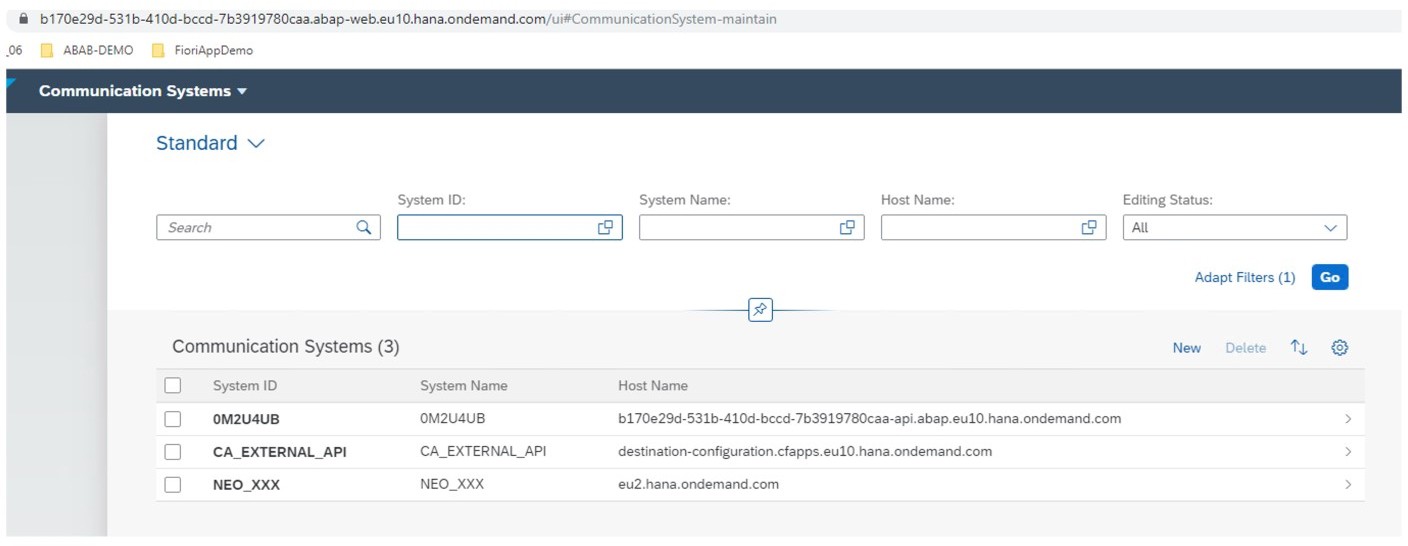
Note:

This information is subject to change. in case of issues, refer to the SSG.

1. In the SAP S/4HANA Dashbord choose *Home* → *Communication Management* → *Communication Systems*.



1. Show the NEO\_XXX Communication Systems:



1. Further Explanations.

a) If there are questions show and explain the following screen:

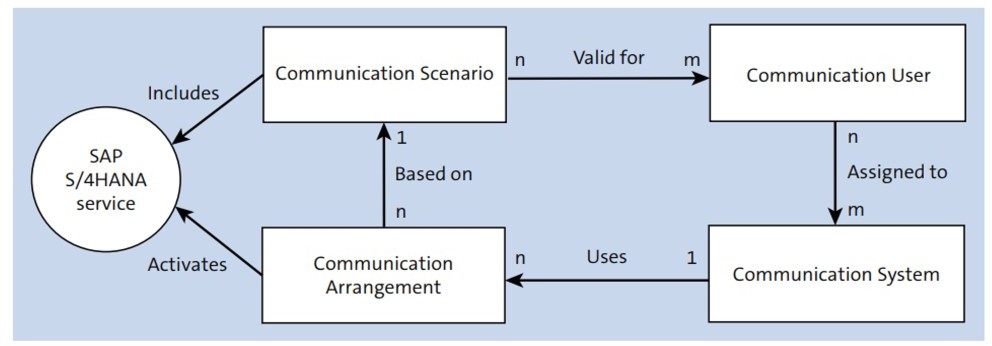


Figure I-3: Data Flow

Communication arrangements

A communication arrangement acts as the primary integration point for an SAP S/4HANA Cloud system and allows you to configure inbound and outbound connections on a semantic level. A communication arrangement bundles a communication scenario and a communication system together and specifies necessary connection parameters such as authentication method and communication user information.

Communication systems

A communication system represents the SAP Cloud Platform subaccount through which you’ll consume your cloud application. The communication system holds technical information such as hostname, IP address, user identity, certificates, etc.

Communication scenarios (also known as integration scenarios)

A communication scenario is the basis on which communication arrangements are configured. Delivered by SAP, communication scenarios bundle services exposed by SAP S/4HANA Cloud. As a user, you can also create your own custom communication scenarios.

LESSON SUMMARY

You should now be able to:

* Explore the most important components of the ABAP installation

Unit 3

Lesson 5

Explaining KYMA

## 87

LESSON OBJECTIVES

After completing this lesson, you will be able to:

* Explain, what Kyma is, what components Kyma is made of and what you can do with it

### Kyma

In particular, we'll look at the following topics in this lesson:

* Kyma in a Nutshell
* Main features
* Key components

Kyma in a Nutshell Kyma:

* is an open-source project designed natively on Kubernetes. It allows you to extend enterprise applications in a quick and modern way, using serverless computing or microservice architecture.
* is avaiaible as a runtime within the SAP Cloud Platform

Kyma allows you to extend applications with microservices and Functions. First, connect your application to a Kubernetes cluster and expose the application's API or events securely. Then, implement the business logic you require by creating microservices or Functions and triggering them to react to particular events or calls to your application's API. To limit the time spent on coding, use the built-in cloud services from Service Catalog, exposed by open service brokers from such cloud providers as GCP, Azure, and AWS.

Kyma comes equipped with these out-of-the-box functionalities:

* Service-to-service communication and proxying (Istio-based Service Mesh).
* Built-in monitoring, tracing, and logging (Grafana, Prometheus, Jaeger, Loki, Kiali).
* Secure authentication and authorization (Dex, Ory, Service Identity, TLS, Role Based Access Control).
* The catalog of services to choose from (Service Catalog, Service Brokers.
* The development platform to run lightweight Functions in a cost-efficient and scalable way (Serverless).
* The endpoint to register Events and APIs of external applications (Application Connector).
* Secure API exposure (API Gateway).
* The messaging channel to receive Events, enrich them, and trigger business flows using Functions or services (Event Mesh, NATS).
* CLI supported by the intuitive UI (Console).
* Asset management and storing tool (Rafter, MinIO).
* Backup of Kyma clusters (Kyma Backup).

Main features

Major open-source and cloud-native projects, such as Istio, NATS, Serverless, and Prometheus, constitute the cornerstone of Kyma. Its uniqueness, however, lies in the "glue" that holds these components together. Kyma collects those cutting-edge solutions in one place and combines them with the in-house developed features that allow you to connect and extend your enterprise applications easily and intuitively.

Kyma allows you to extend and customize the functionality of your products in a quick and modern way, using serverless computing or microservice architecture.

The extensions and customizations you create within Kyma are decoupled from the core applications, which means that:

* Deployments are quick.
* Scaling is independent from the core applications.
* The changes you make can be easily reverted without causing downtime of the production system.

Last but not least, Kyma is highly cost-efficient. All Kyma native components and the connected open-source tools are written in Go. It ensures low memory consumption and reduced maintenance costs compared to applications written in other programming languages such as Java.

Key components

Kyma is built of numerous components but these three drive it forward:

* Application Connector
* Serverless
* Service Catalog

Further details:

Application Connector:

* + Simplifies and secures the connection between external systems and Kyma
  + Registers external Events and APIs in the Service Catalog and simplifies the API usage
  + Provides asynchronous communication with services and Functions deployed in Kyma through Events
  + Manages secure access to external systems
  + Provides monitoring and tracing capabilities to facilitate operational aspects Serverless:
    - Ensures quick deployments following a Function approach
    - Enables scaling independent of the core applications
    - Gives a possibility to revert changes without causing production system downtime
    - Supports the complete asynchronous programming model
    - Offers loose coupling of Event providers and consumers
    - Enables flexible application scalability and availability Service Catalog
    - Connects services from external sources
    - Unifies the consumption of internal and external services thanks to compliance with the Open Service Broker standard
    - Provides a standardized approach to managing the API consumption and access
    - Eases the development effort by providing a catalog of API and Event documentation to support automatic client code generation

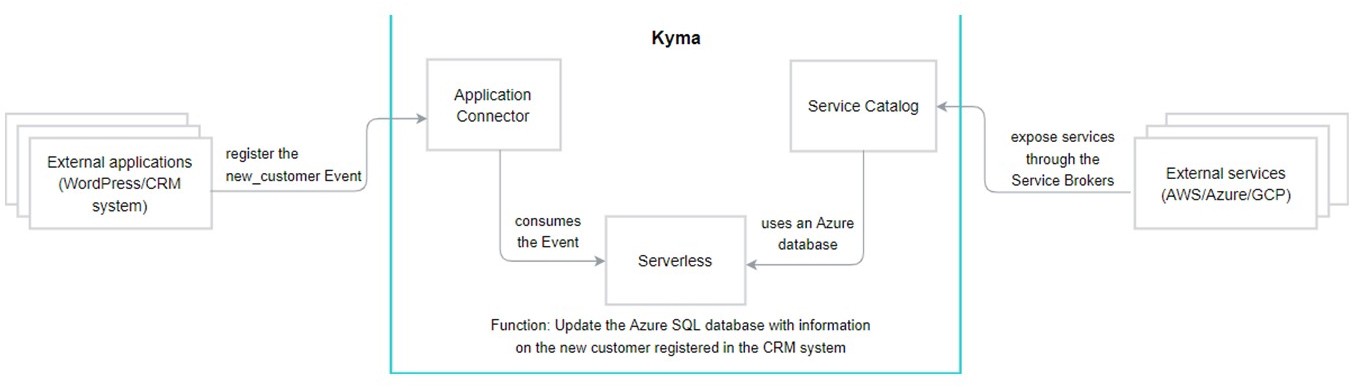
This basic use case shows how the three components work together in Kyma:



Figure 54: The Main Components Works Together

On the left side is the application, here Wordpress. On the right side the available services of the Hyperscaler. In the middle you see the principle of the application connector which connects the app to be extended with the platform. The serverless runtime which contains the code that is extended by various services from the service catalogue.



Figure 55: Kyma as Runtime at SAP Cloud Platform

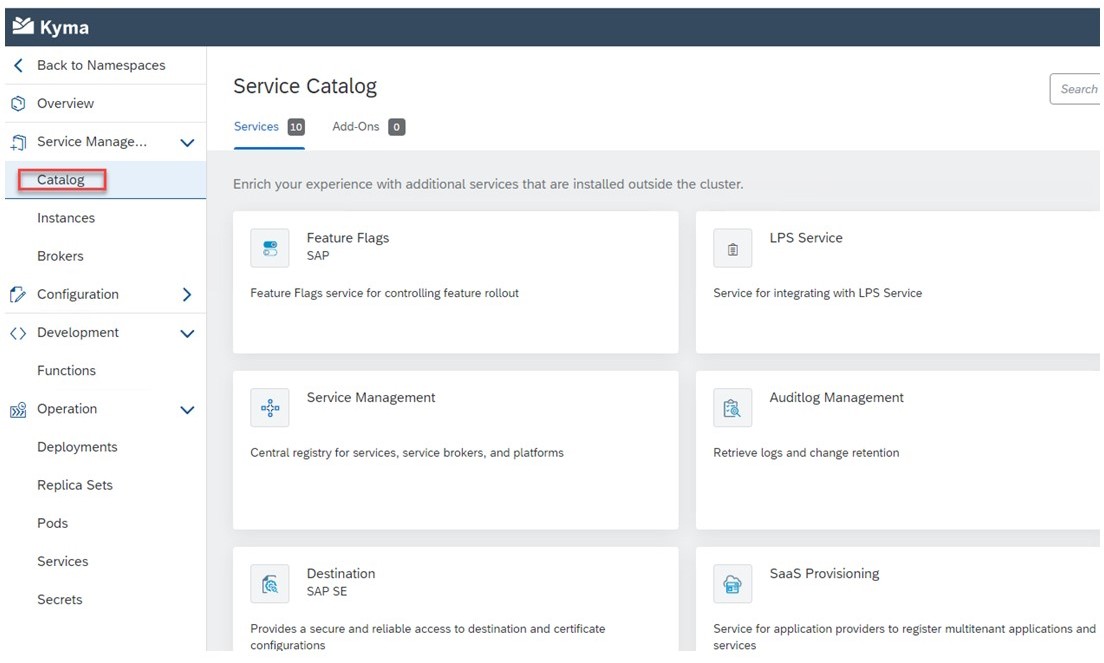
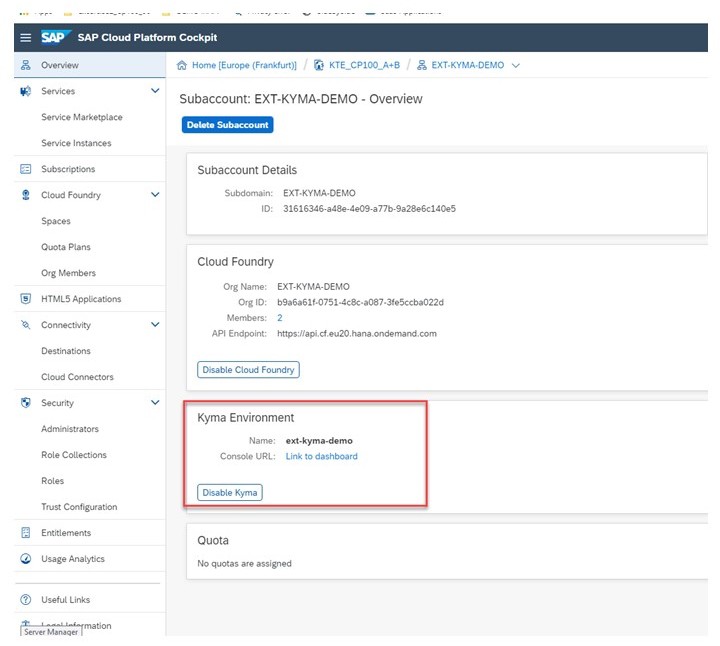
The figure shows how the Kyma runtime is activated at SAP Cloud Platform.



Figure 56: Service Catalog

Within the service catalog in KYMA.

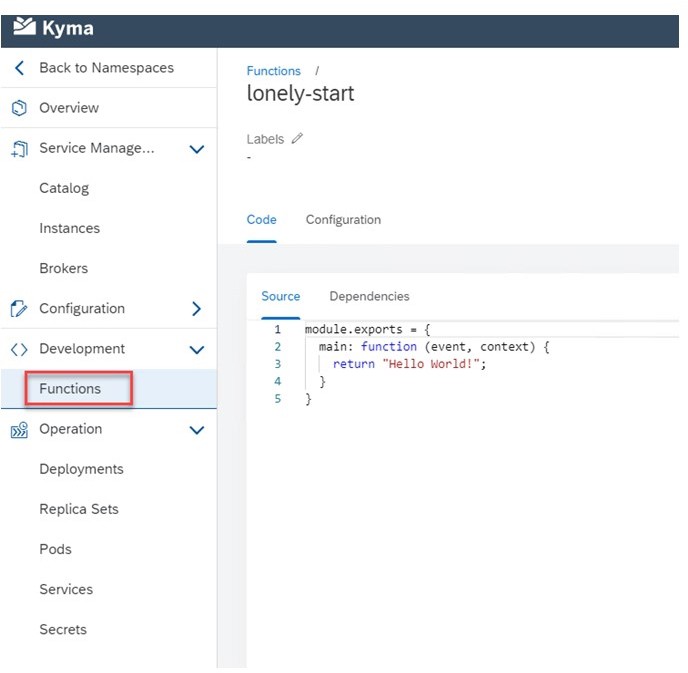


Figure 57: Serverless and Functions

Functions in action.

How to Explore Kyma

1.

LESSON SUMMARY

You should now be able to:

* Explain, what Kyma is, what components Kyma is made of and what you can do with it