



Develop Locally with SAP HANA, Express Edition, and Deploy to Cloud Foundry

CPL166

Exercises / Solutions

Andreas Haberl/Christian Paulus/Johannes Scheerer/Stephen Cherian/Jain Basil Aliyas/Sven
Kohlhaas, SAP
Q3/Q4, 2017

TABLE OF CONTENTS

SCENARIO OVERVIEW3

WHAT YOU WILL LEARN3

EXERCISES4

Exercise 1: Create Project and Database Artifacts4

Exercise 2: Adding Business Logic with JavaScript4

Exercise 3: Adding a User Interface4

Exercise 4: Adding Authentication and Authorization to your Application4

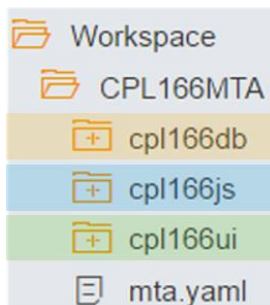
Exercise 5: Deploying your Multi Target Application to Cloud Platform4

Scenario Overview

With SAP HANA 1 SPS 11, SAP has introduced a major new application server architecture: Extended Application Services for SAP HANA, advanced model (XS Advanced) on premise. This architecture is based on Cloud Foundry, which is now also available in SAP Cloud Platform. In this session, you are going to develop an application using SAP HANA, express edition, and SAP Web IDE locally; then deploy the application to Cloud Foundry with SAP HANA on SAP Cloud Platform. The application will be built using Microservices and leverages the powerful HANA database capabilities in combination with flexible and scalable cloud concepts.

With the programming model of XS Advanced you are not bound to a specific programming language. The platform itself can be extended with support for a broad variety of languages – just like in Cloud Foundry. In our scenario, we will focus on HANA database development, application logic using JavaScript and Node.js plus a User Interface using SAPUI5 backed by an OData Service you will expose. To introduce security aspects in your application, we will make use of the User Authorization and Authentication backing service (UAA) available on both XS Advanced and Cloud Foundry.

To ensure proper configuration and deployment of the services we are going to develop, we will use the Multi Target Application descriptor (mta.yaml) - a platform independent description of the different application modules, their interdependencies and configuration data they expose, and the resources they require to run. Using this descriptor, it is ensured that the application can be deployed on both XS Advanced on premise systems and Cloud Foundry.



Structure of a Multi Target Application Project with database module, Node.js module and html5 user interface module

WHAT YOU WILL LEARN

In this hands-on session you will learn:

- to create Multi Target Application projects in SAP WebIDE
- to create HANA database artifacts using HANA Deployment Infrastructure and Core Data Services
- to create HANA business logic using SQL procedures
- to create business logic using Node.js
- to build a user interface backed by an OData service
- to use the UAA backing service to ensure security for your application
- build your application and deploy to Cloud Foundry

EXERCISES

This hands-on session consists of following exercises:

Exercise 1: Create Project and Database Artifacts

In this exercise, you will use SAP Web IDE on HANA to create a Multi Target Application Project with a database module.

Follow the link to GitHub.com to get to the step-by-step description of this exercise:

<https://github.com/SAP/cloud-xsa-hybrid-deployment-tutorial/exercise1>

Exercise 2: Adding Business Logic with JavaScript

In this exercise, you will create and test a Node.js module with additional business logic and expose an OData service which we can consume in a user interface later.

Follow the link to GitHub.com to get to the step-by-step description of this exercise:

<https://github.com/SAP/cloud-xsa-hybrid-deployment-tutorial/exercise2>

Exercise 3: Adding a User Interface

In this exercise, you will create and test the user interface for your application by using the OData service we created during exercise 2 in SAPUI5.

Follow the link to GitHub.com to get to the step-by-step description of this exercise:

<https://github.com/SAP/cloud-xsa-hybrid-deployment-tutorial/exercise3>

Exercise 4: Adding Authentication and Authorization to your Application

In this exercise, you will add authentication and authorization using the UAA service. Authentication forces users to log on and identify themselves, before they can use the application. You will also add authorization checking code to introduce two categories of users.

Follow the link to GitHub.com to get to the step-by-step description of this exercise:

<https://github.com/SAP/cloud-xsa-hybrid-deployment-tutorial/exercise4>

Exercise 5: Deploying your Multi Target Application to Cloud Platform

In this exercise, you will create the MTA archive as the deployable unit using the SAP Web IDE export functionality and deploy it to the Cloud Platform trial landscape using the Cloud Foundry command line tool.

Follow the link to GitHub.com to get to the step-by-step description of this exercise:

<https://github.com/SAP/cloud-xsa-hybrid-deployment-tutorial/exercise5>

www.sap.com/contactsap

© 2017 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company.

The information contained herein may be changed without prior notice. Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

In particular, SAP SE or its affiliated companies have no obligation to pursue any course of business outlined in this document or any related presentation, or to develop or release any functionality mentioned therein. This document, or any related presentation, and SAP SE's or its affiliated companies' strategy and possible future developments, products, and/or platform directions and functionality are all subject to change and may be changed by SAP SE or its affiliated companies at any time for any reason without notice. The information in this document is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, and they should not be relied upon in making purchasing decisions.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies. See <http://www.sap.com/corporate-en/legal/copyright/index.epx> for additional trademark information and notices.