

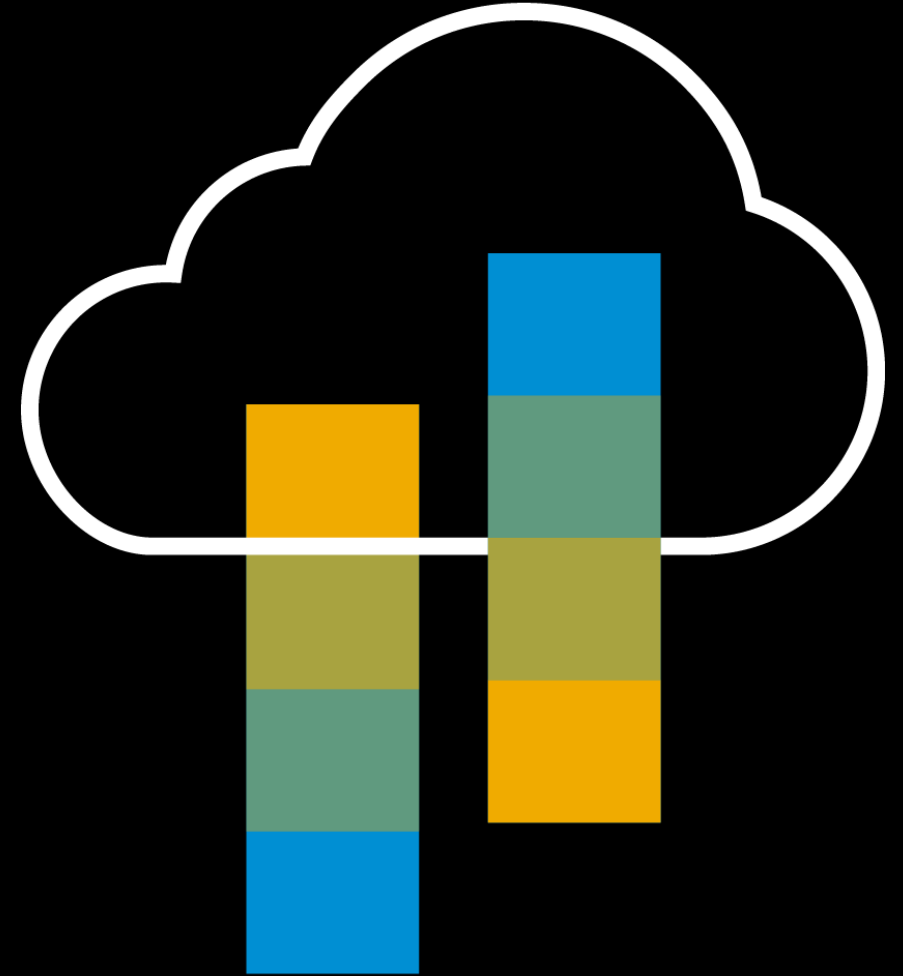
# SAP S/4HANA Cloud SDK:

## Develop your first SAP S/4HANA Extension App on SAP Cloud Platform Cloud Foundry

October 13, 2017

Henning Heitkötter, SAP S/4HANA Cloud SDK

PUBLIC



# Disclaimer

The information in this presentation is confidential and proprietary to SAP and may not be disclosed without the permission of SAP. Except for your obligation to protect confidential information, this presentation is not subject to your license agreement or any other service or subscription agreement with SAP. SAP has no obligation to pursue any course of business outlined in this presentation or any related document, or to develop or release any functionality mentioned therein.

This presentation, or any related document and SAP's strategy and possible future developments, products and or platforms directions and functionality are all subject to change and may be changed by SAP at any time for any reason without notice. The information in this presentation is not a commitment, promise or legal obligation to deliver any material, code or functionality. This presentation is provided without a warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement. This presentation is for informational purposes and may not be incorporated into a contract. SAP assumes no responsibility for errors or omissions in this presentation, except if such damages were caused by SAP's intentional or gross negligence.

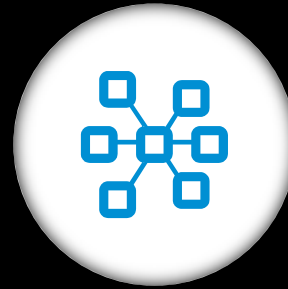
All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, which speak only as of their dates, and they should not be relied upon in making purchasing decisions.

**This meeting will be recorded.**

# Agenda



**DEVELOPMENT OF  
CLOUD-NATIVE  
APPLICATIONS**



**INTRODUCTION TO  
THE SAP S/4HANA  
CLOUD SDK**



**WALKTHROUGH**

# Cloud-native Applications

# Side-by-side Extensibility of SAP S/4HANA Cloud

## Smart App Development

### Digital transformation requires



Differentiating and responding quickly to changing market conditions

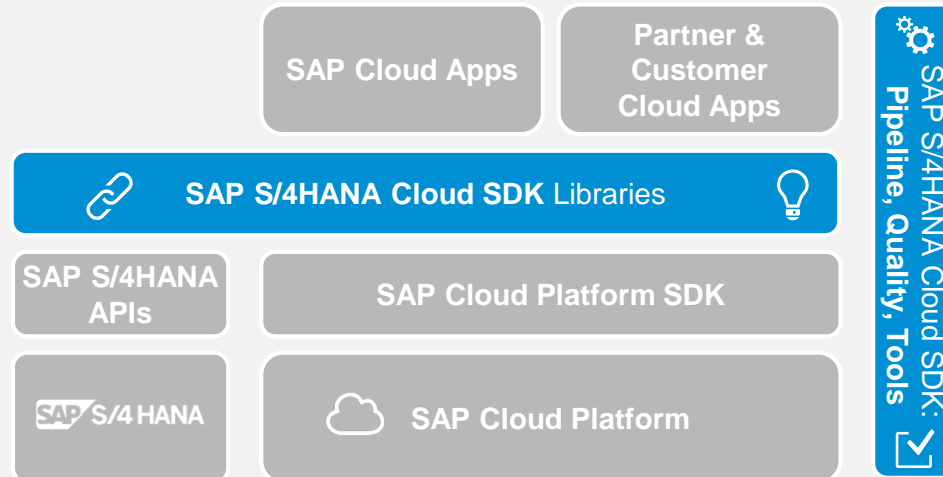


Best-in-class software quality to mitigate risks introduced by change



Ensuring fast time-to-value by leveraging rich ecosystem

### SAP S/4HANA Cloud Software Development Kit (SDK)



### SAP S/4HANA Cloud SDK provides



Boost of development efficiency for flexible innovations on top of SAP S/4HANA functionality on SAP Cloud Platform



Established and proven quality standards ensured by automated means of SAP S/4HANA Cloud SDK



SAP App Center as Go-to-Market platform for all digital services by SAP and partners

# Engineering principles for successful cloud development



## SCALABILITY

Design applications for horizontal scalability.

Partition and decompose workloads into discrete units.

Ensure applications and services are stateless.

Cache items that do not change much.



## RESILIENCE

Understand potential failures, impact, and recovery.

Use load balancing to distribute workloads.

Use bulkheads to contain potential failures.

Use circuit breakers to handle persistent failures.



## SECURITY

Don't trust, verify.

Apply defense in depth, secure all resources.

Fail securely.

Protect data at rest.

Use only secure tunnels for on-prem connectivity.



## DESIGN

Prefer loosely-coupled components with asynchronous coms.

Separate infrastructure logic from domain logic.

Prefer REST APIs for external communication.

Prefer asynchronous messaging for internal coms.



## OPERATIONS

Design for IT ops (deploy, monitor, investigate, secure)

Automate build and deploy processes.

Implement logging and alerting in all components.

Inventory, inspect and audit cloud resources.

These engineering principles are reflected in the **twelve-factor app** methodology for building software-as-a-service apps:  
<https://12factor.net/>

**SAP S/4HANA Cloud SDK**

# Introduction

## SAP S/4HANA Cloud SDK

SAP S/4HANA Cloud SDK reduces the effort of building applications running on SAP Cloud Platform for extending S/4HANA, by providing Java libraries and toolsets for developers



**Goal: attractive development environment** to enable customers and partners to efficiently develop applications for S/4HANA on SAP Cloud Platform



**Framework of choice** for extending S/4HANA (Cloud & on premise) on SAP Cloud Platform (Neo & Cloud Foundry)



**Helps ensuring high S/4HANA quality standards** with regards to performance, resource consumption and operations also on SAP Cloud Platform



# Value Proposition

## SAP S/4HANA Cloud SDK

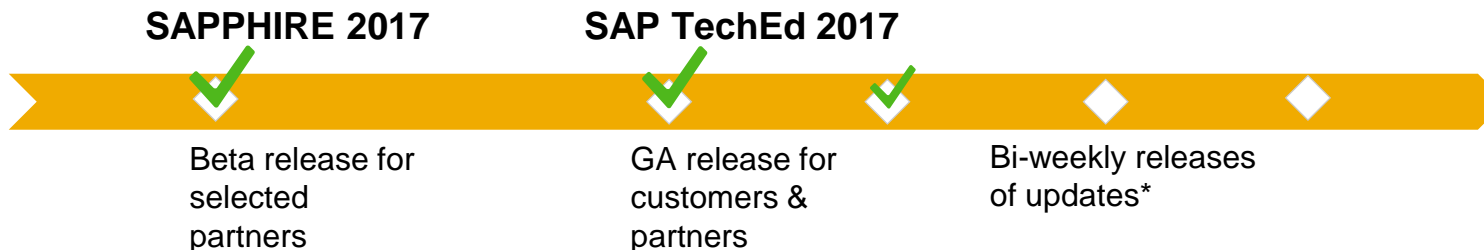
### Key Objectives

- Provide **attractive development environment** to enable customers and partners to efficiently develop applications for S/4HANA on SAP Cloud Platform
- **Ensure high S/4HANA quality standards** with regards to performance, resource consumption and operations

### Key Features of SAP S/4HANA Cloud SDK

- ▶ **Enable developers to easily connect to S/4HANA systems** while leaving the configuration to SAP Cloud Platform: S/4HANA Virtual Data Model & abstract layer for connections to S/4HANA Cloud and on premise systems
- ▶ **Facilitate effective coding** and support seamless transition between SAP CP Neo and SAP CP Cloud Foundry: Abstraction layer for key platform services such as multi-tenancy and authentication
- ▶ **Ensure high quality native cloud software**, e.g. by offering resilience engineering basics out of the box, and by supporting the integration of **third-party libraries** such as state-of-the-art frameworks for logging, data persistency frameworks, feature toggling and others
- ▶ **Jump start application development** by offering project templates, tutorials, and code samples
- ▶ **Continuous integration and delivery pipeline** provided out-of-the-box, including testing support and code checks

### Timeline



\*based on current planning / subject to change

# Walkthrough

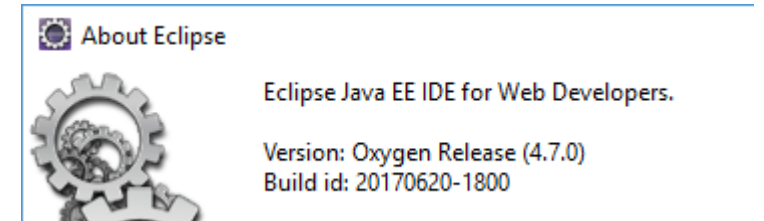
# Overview

- A. Generate Project & Explore Structure
  - B. Package App & Deploy on Cloud Foundry
  - C. Extend App: Integrate with S/4HANA
  - D. Make App Resilient
  - E. Implement Caching
- 
- Walkthrough covers steps 1–6 & 10 of SAP S/4HANA Cloud SDK Tutorial Series (example adapted, see [www.sap.com/s4sdk](https://www.sap.com/s4sdk) > Community > SAP S/4HANA Cloud SDK Overview – [direct link](#))

# Local Environment Setup

- Java (JDK, 1.8.0)
  - Java home: points to JDK
- Apache Maven (3.5.0)
- Eclipse (Oxygen Release) Java EE IDE for Web Developers (optional)
- Cloud Foundry Command Line Interface (CLI; optional)

```
~\Documents\git-repos> mvn -v
Apache Maven 3.5.0 (ff8f5e7444045639af65f6095c62210b5713f426; 2017-04-03T21:39:06+02:00)
Maven home: C:\Program Files\Java\apache-maven-3.5.0\bin\..
Java version: 1.8.0_72, vendor: Oracle Corporation
Java home: C:\Program Files\Java\jdk1.8.0_72\jre
Default locale: en_US, platform encoding: Cp1252
OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"
```



```
~\Documents\git-repos> cf -v
cf.exe version 6.26.0+9c9a261.2017-04-06
```

# Setup Instructions

- Install JDK8
- Install Maven
- (optional) Install the Cloud Foundry command line interface (CLI) [see below]

How? You can either follow [step 1 of our tutorial series](#) (which is using a convenient package manager) or follow the description below to install the tools manually.

## Install JDK8

- [Download JDK 8](#) and install it
- Windows:
  - Run the console cmd and enter `setx JAVA_HOME "C:\Program Files\Java\jdk1.8.0_X"` (replace the path with the path leading to your JDK installation)
  - To test your installation, open a new console and run `java -version`. This command should return "java version 1.8.0\_X".
- MacOSX: nothing to do, env variables should be adjusted automatically.

## Install Maven

- [Download the latest release](#) (3.5.0) as binary zip archive
- [Follow the description on this web page to install Maven](#)
- Test by running `mvn -version` on the console. This should return "Apache Maven 3.5.0" and some more information about your system.

(optional) Install the Cloud Foundry command line interface (CLI)

- Follow the description in the section [Setup for Cloud Foundry](#)

# A) Generate Project & Explore Structure

## Tutorial: Step 3

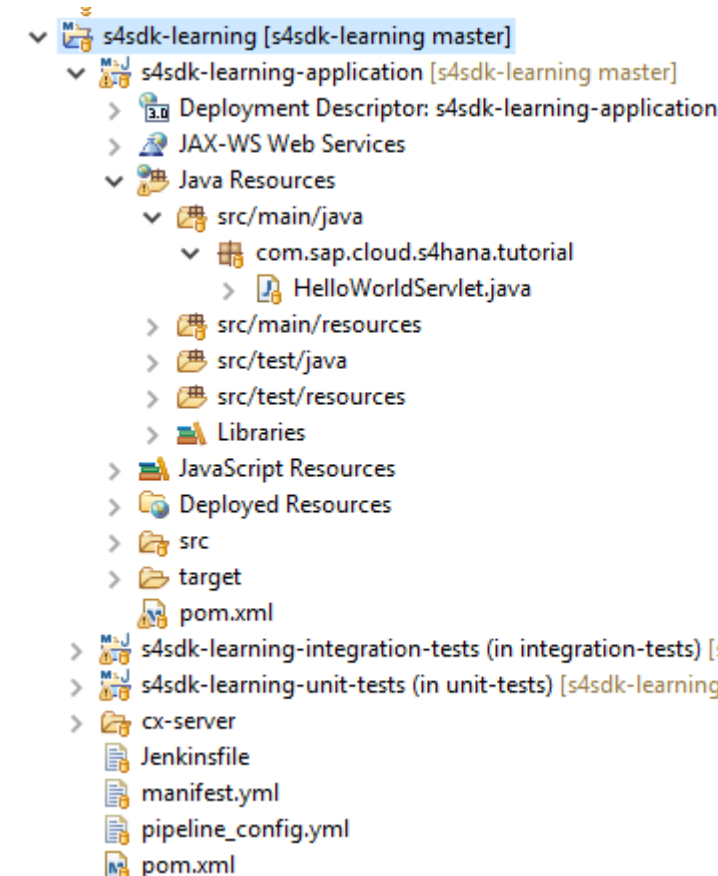
### Run

- `mvn archetype:generate -DarchetypeGroupId=com.sap.cloud.s4hana.archetypes -DarchetypeArtifactId=scp-cf-tomee -DarchetypeVersion=1.1.2`
  - groupId: `com.sap.cloud.s4hana.tutorial`
  - artifactId: `s4sdk-learning`
  - version and package: keep defaults (hit enter)
  - uniqueHostname: `s4sdk-learning-<unique-identifier>`

### Import generated Maven Projects into Eclipse

### Explore generated folder `s4sdk-learning`

- Take a look at  
`./application/src/main/java/com/sap/cloud/s4hana/tutorial/HelloWorldServlet.java`



## B) Package App & Deploy on Cloud Foundry

[Tutorial: Step 3](#)

### Run

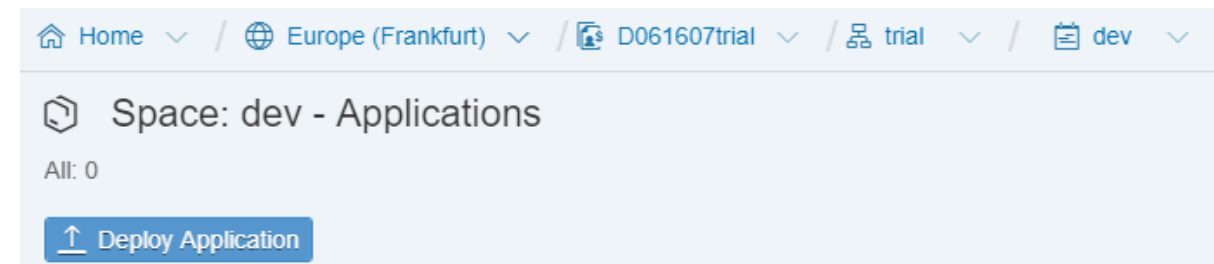
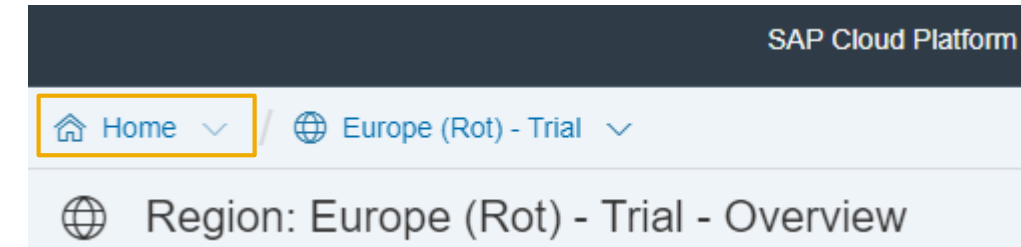
- `cd s4sdk-learning`
- `mvn clean package`

### Deploy to Cloud Platform Cloud Foundry

1. Visit <https://cloudplatform.sap.com/try.html> and login
2. In Cloud Platform Cockpit, go to “Home” and create Cloud Foundry trial in region Europe (Frankfurt)
3. Go to trial > Spaces > dev
4. Choose Deploy Application
  - File Location: `./application/target/s4sdk-learning-application.war`
  - Use Manifest & Location: `./manifest.yml`

– OR run the following with CF CLI

  - `cf api https://api.cf.eu10.hana.ondemand.com`
  - `cf login`
  - `cf push`
5. Visit <https://your-unique-hostname.cfapps.eu10.hana.ondemand.com/hello>



## C) Extend App: Integrate with S/4HANA

[Tutorial: Step 4](#)  
& [Step 10](#)

### Create *BusinessPartnerServlet*

- Copy & paste [template](#) to  
`./application/src/main/java/com/sap/cloud/s4hana/tutorial/BusinessPartnerServlet.java`
- Search for representation of business partner service in package  
`com.sap.cloud.sdk.s4hana.datamodel.odata.services` and call method to get all business partners
- Select the fields business partner (ID) and business partner name
- Filter for organizations (business partner category “2”)
- Execute the service call
- Return the result as JSON using `new Gson().toJson(...)`



## C) Extend App: Test Integration with S/4HANA

[Tutorial: Step 4](#)  
& [Step 10](#)

### Add test

- Add [./integration-tests/src/test/java/com/sap/cloud/s4hana/tutorial/BusinessParterServiceTest.java](#)

### Prepare test environment

- Add [./integration-tests/src/test/resources/systems.json](#) and adapt to your S/4HANA Cloud system
- Add `./integration-tests/src/test/resources/credentials.yml` based on [template](#) with the credentials of your communication user

### Run test

- `mvn clean package` will also run the new test

## C) Extend App: Deploy Integration with S/4HANA

[Tutorial: Step 4](#)  
& [Step 10](#)

### Deploy on Cloud Foundry

- `mvn clean package`
- Deploy again (as before)
- Set environment variable destinations to the following value
  - [{name: 'ErpQueryEndpoint', url: 'https://URL-to-S4HANA', username: '...', password: '...'}]
  - Go to app > User-provided variables > Add variable
  - OR `cf set-env s4sdk-learning destinations "[{name: 'ErpQueryEndpoint', url: 'https://URL-to-S4HANA', username: '...', password: '...'}]"`
- Restart app (> Overview > Restart OR `cf restart s4sdk-learning`)

## D) Make App Resilient

### Introduce Hystrix command

- Copy & paste [template](#) to  
`./application/src/main/java/com/sap/cloud/s4hana/tutorial/GetBusinessPartnersCommand.java`
- Implement run method based on previous step
  - Command shall wrap the call to service and return result
  - Use `getConfigContext()` to retrieve ERP context
- Adapt `BusinessPartnerServlet`: create command (with ERP context) and execute
- Test

### Deploy on Cloud Foundry

- `mvn clean package`
- Deploy again (as before)
- Set environment variable `destinations` as before (not necessary with `cf push`)
- Add environment variable `ALLOW MOCKED AUTH HEADER` with value `true`
- Restart app

## E) Implement Caching

[Tutorial: Step 6](#)  
& [Step 10](#)

Adapt *GetBusinessPartnersCommand* to cache the result

- Extend *CachingErpCommand*
- Rename `run()` method to `runCacheable()`
- Create cache field and initialize with `com.google.common.cache.CacheBuilder.newBuilder().build();`
- Implement `getCache()` method from super class and return cache

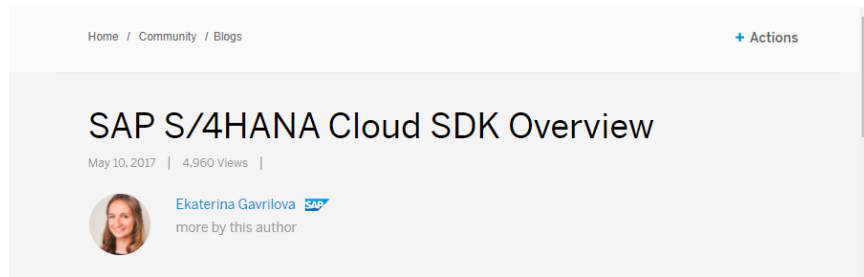
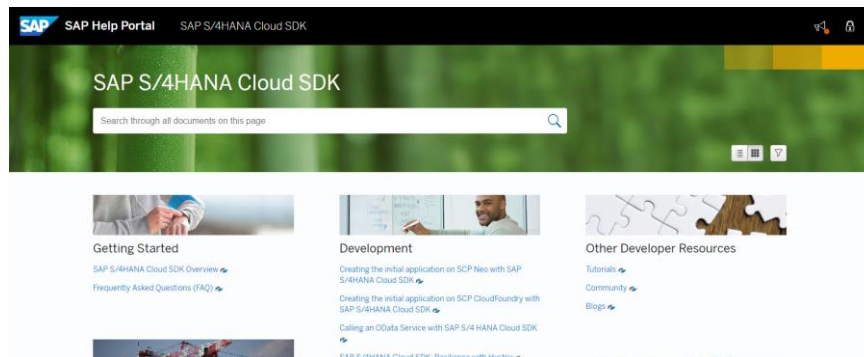
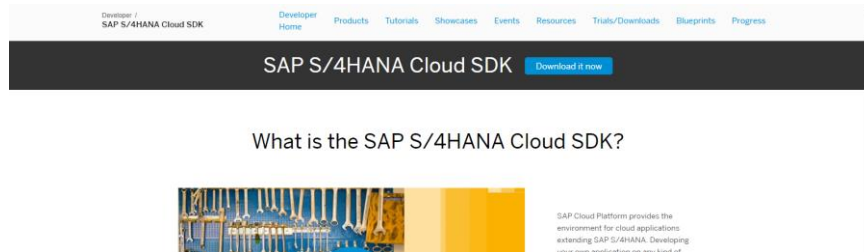
# Troubleshooting

- Add the following to ./integration-tests/pom.xml

```
<dependency>  
  <groupId>ch.qos.logback</groupId>  
  <artifactId>logback-classic</artifactId>  
  <version>1.2.3</version>  
  <scope>test</scope>  
</dependency>
```

# Information Resources

## SAP S/4HANA Cloud SDK



Official website of SAP S/4HANA Cloud SDK on sap.com

[www.sap.com/s4sdk](http://www.sap.com/s4sdk)

Official help page

[help.sap.com](http://help.sap.com) > [S/4HANA Cloud SDK](#)

... including tutorials

[blogs.sap.com](http://blogs.sap.com)

Component:

XX-S4C-SDK

Central contact address:

[s4sdk@sap.com](mailto:s4sdk@sap.com)

# Developing extensions for S/4HANA using SAP S/4HANA Cloud SDK

## Step 1

Enroll to SAP PartnerEdge program



### Enroll

Our [engagement model](#) was created for partners that design, develop, and build applications, software, and integrated solutions based on SAP technology and platforms.

If you're an ISV, developer or even looking to bundle or embed SAP technologies with your own via an OEM relationship – you're in the right place...

## Step 2

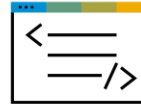
Build your S/4HANA solution



### Discover

Leverage your individual expertise to design extensions for SAP S/4HANA.

- Get started with [dev enablement content](#) and application architecture concepts
- Register your idea with SAP and run a launch workshop with SAP



### Develop

Develop an extension for SAP S/4HANA on SAP Cloud Platform using SAP S/4HANA Cloud SDK

- Get your S/4HANA Cloud Innovation Pack from [SAP PartnerEdge](#)
- Download the [SAP S/4HANA Cloud SDK](#)
- Setup your [development environment](#)
- Get started with [the development tutorial](#)



### Deploy

Deploy your S/4HANA extension in your SAP Cloud Platform environment

- Publish your extension to your [SAP Cloud Platform](#) tenant
- Run your extension against your development SAP S/4HANA system

## Step 3

Market your S/4HANA extension



### SAP App Center

The [SAP App Center](#) is a globally available, multi-currency digital marketplace for enterprise applications that extend the SAP digital core.

SAP App Center provides a role based workflow driven platform to manage the entire lifecycle of purchases, from license acquisition to user management and renewals.

# Thank you.

Contact information:

**Henning Heitkötter**

[s4sdk@sap.com](mailto:s4sdk@sap.com)



© 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://global.sap.com/corporate-en/legal/copyright/index.epx> for additional trademark information and notices.