

Session 4.

Oracle Developer Cloud

Ram N Sangwan



Agenda



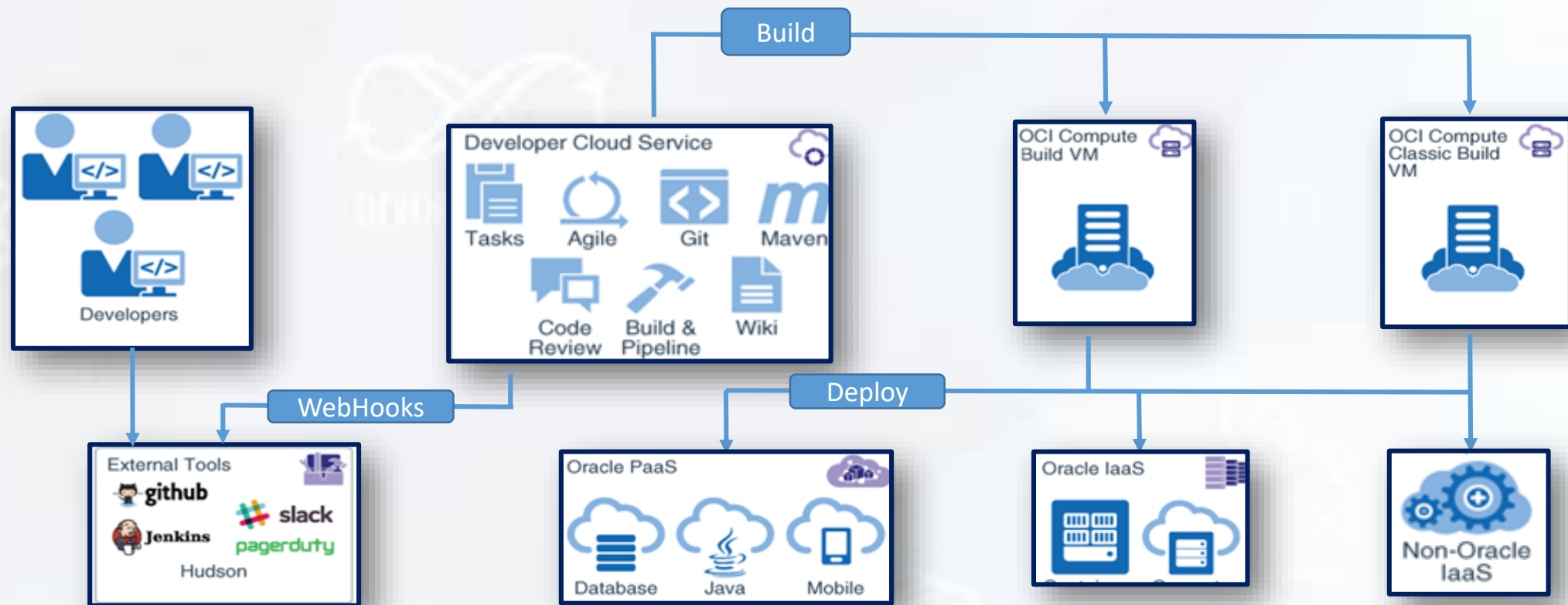
- Key Components of Oracle Developer Cloud
- Using an IDE with Oracle Cloud
- Eclipse Overview
- Support for IDE Integration with DevCs
- Build Process & Project Management



Oracle Developer Cloud Service



- A cloud-based software development Platform as a Service (PaaS) and a hosted environment for your application development infrastructure.
- It provides an open source standards-based solution to plan, develop, collaborate, build, and deploy applications in Oracle Cloud.





DevCS Features



- **Git repositories** – To host your application source code files and track their versions
- **Maven repositories** – To host library and binary dependencies of your Maven applications
- **Build system** – To automate builds with CI and CT using OCI Compute VMs
- **Deploy system** - Automate deployment to Oracle Java Cloud Service.
- **Code review** - peer review the code updates
- **Issue tracking system**, so you can track tasks, defects, and features
- **Wikis**, so you can collaborate with your team
- Integration with Eclipse IDE, Oracle JDeveloper, and NetBeans IDE
- Integration with external software using webhooks



Key Components



Term/components	Description
Applications	Oracle Cloud Applications are a set of modular Cloud-ready enterprise applications.
Organization	The top-most entity in the project structure of DevCS.
Git repository	A Source Code Management (SCM) and distributed version control tool to host source code files.
Maven repository	A hosted repository to store build artifacts, library files, and dependencies for Maven applications.
Issue tracker	A built-in issue management system to create and track tasks, defects, and features.
Merge Request and Code review	A method to merge a Git repository branch with another branch. Before merging the branches, team members can review differences
Build System	A built-in system to define and automate builds of your applications.



Key Components



Job	A configuration that defines your application's builds.
Build	The result from a job's run.
Pipeline	A path or a chain of builds. A pipeline helps you run continuous integration jobs and reduce network traffic.
Wiki	Built-in wiki system to help your team author and manage wiki pages.
Deployment configuration	A configuration to deploy a build artefact to an Oracle Java Cloud Service or Oracle Java Cloud Service - SaaS Extension instance.
Oracle Cloud Infrastructure	Oracle Cloud Infrastructure is a set of cloud services that enable you to build and run a wide range of applications and services in a highly available hosted environment.
Oracle Java Cloud Service	Oracle Cloud service to deploy web applications to a public Oracle WebLogic Server domain on Oracle Cloud.



Key Components



Term	Description
Visual Builder	A Visual Builder instance that provides the server for delivering pages in web applications, and services your web and mobile apps might use to access data, including the database used to store data and the proxy server for managing connections to REST services.
Visual application	A responsive web or native mobile application developed using VB Studio's browser-based development environment. You deploy a visual application to a Visual Builder instance.
Project	A collection of VB Studio features. You can use a project to host source code files, track issues, collaborate on code, build, and deploy your applications. A project can host multiple Git repositories.



Key Components



VB Studio Designer	VB Studio's browser-based development environment.
Environment	Defines the target Oracle Cloud Applications, Visual Builder, Oracle Cloud SaaS, or Oracle Cloud Infrastructure service instance as a single entity. You'll define an environment to deploy an application to a service instance.
OCI Object Storage	Oracle Cloud service that hosts containers on Oracle Cloud to store project data. VB Studio uses the containers to archive build artifacts and Maven artifacts, and export project data.
Oracle Application Container Cloud Service	Oracle Cloud service to deploy Java SE, Node.js, PHP, Python, Ruby, Go, Java EE 7 (or later), and .NET applications to Oracle Cloud.



Using an IDE with Oracle Cloud



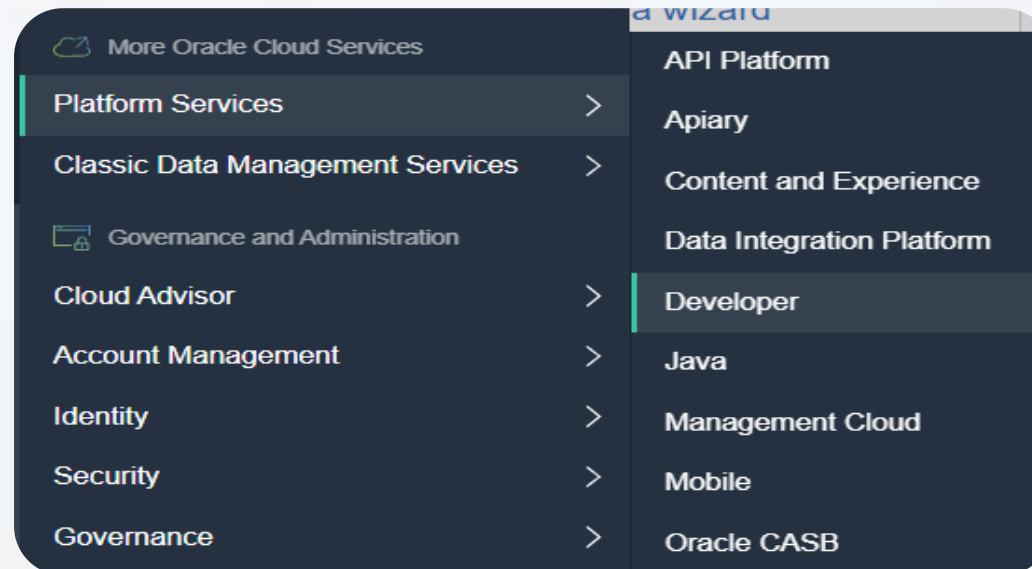
Using the Oracle Developer Cloud Service integration, you can access and update application source code files, manage issues, and monitor builds of your applications from the any of the following supported IDEs.

- **Jdeveloper**
- **NetBeans IDE**
- **Eclipse IDE**



IDE with Oracle Cloud – Pre-Requisite

- Create an Oracle Java Cloud Service instance.
<https://console.us-ashburn-1.oraclecloud.com/>
- Install the IDE on your local machine.
 - If you want to use Eclipse, then also install Oracle Enterprise Pack for Eclipse from – For fresh Eclipse Installation
<http://www.oracle.com/technetwork/developer-tools/eclipse/downloads/index.html>

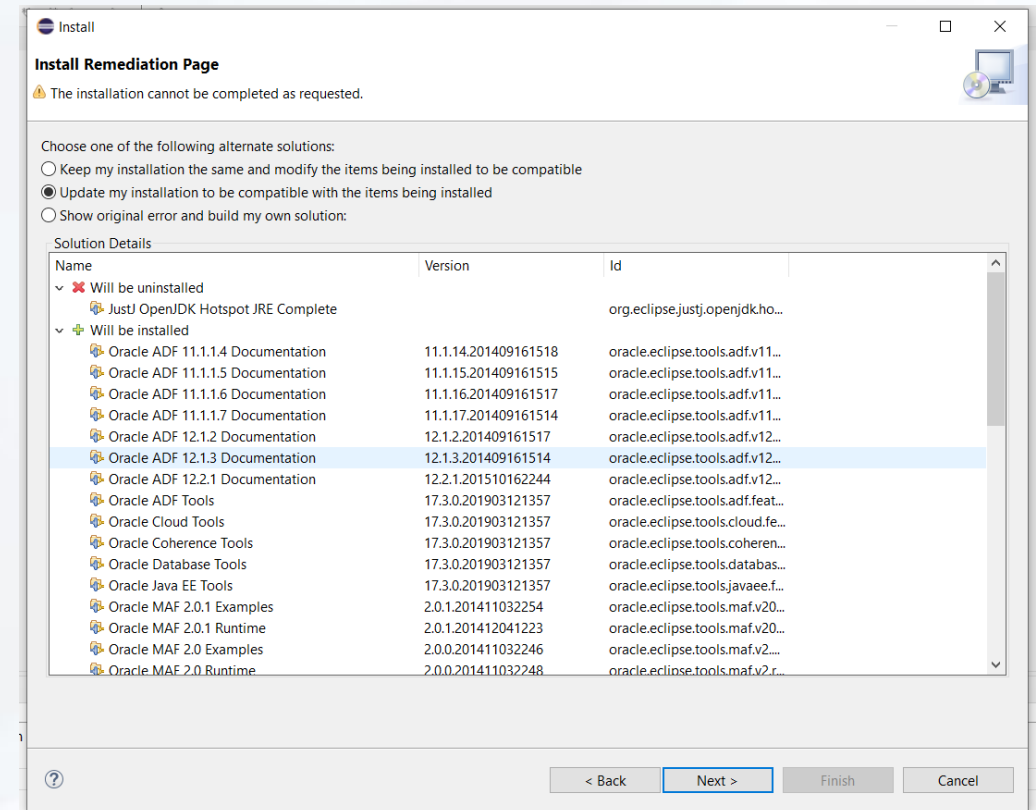
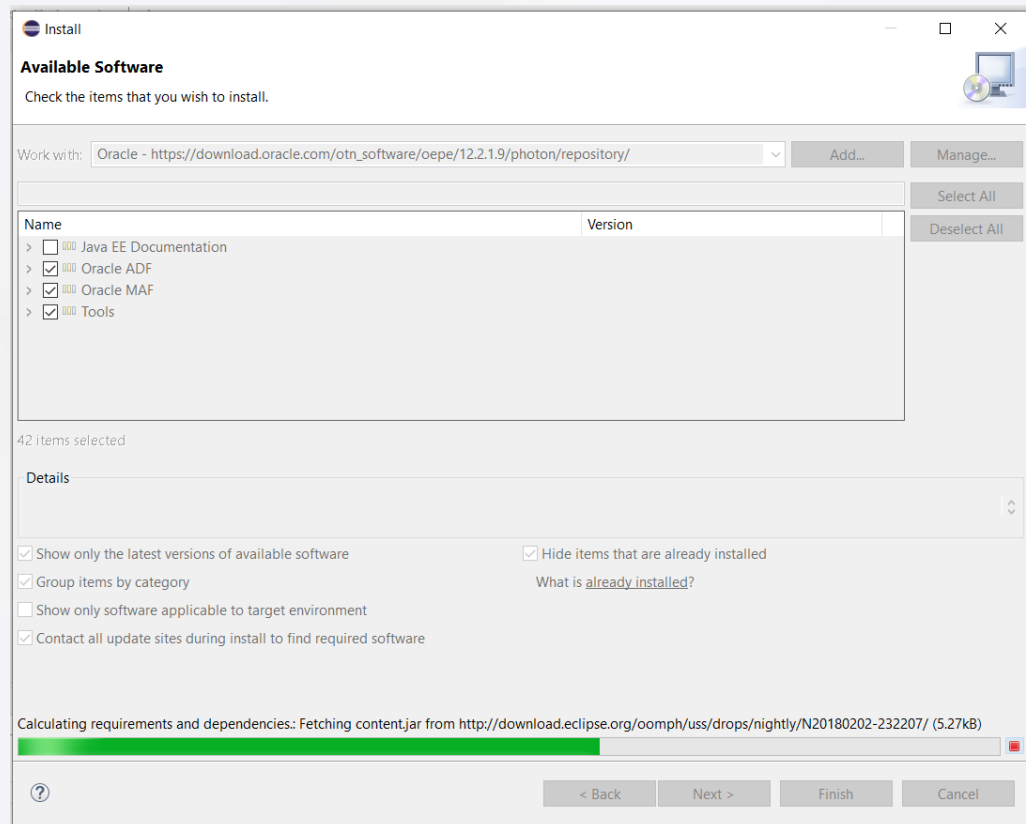




Install on Existing Eclipse

- You can use OEPE plugin repository to install the required Oracle Tools.

http://download.oracle.com/otn_software/oepe/12.2.1.9/photon/repository/





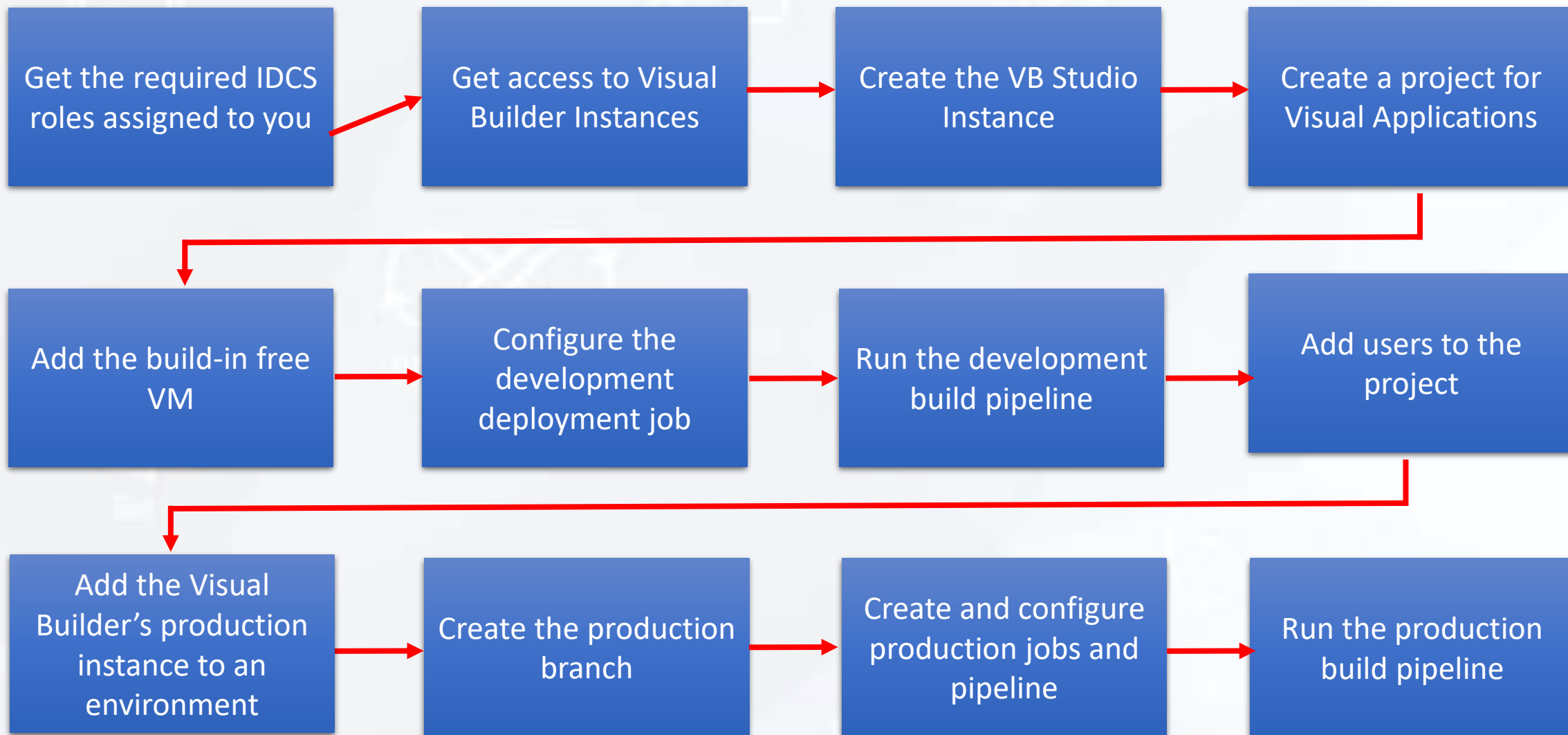
Eclipse IDE



- You can use the Eclipse IDE and the Oracle Enterprise Pack for Eclipse (OEPE) to access Oracle Developer Cloud Service (DevCS) projects, its Git repositories, issues, and builds.
- You use the Oracle Cloud view of the Eclipse IDE to access projects.
- The IDE uses
 - Mylyn (a plugin) to access issues and
 - EGit to access Git repositories.



Steps to set up VB Studio for Visual Applications





Reasons for sequence of Steps

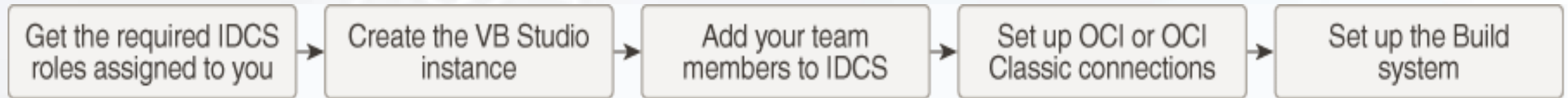


Action	Why?
1. Get the required IDCS roles	To create and set up the VB Studio instance, you must be assigned some specific IDCS roles.
2. Get access to VB instances	To deploy an app to the VB instance, you need a user's credentials. You must create the instance if needed.
3. Create VB Studio instance	To use any Oracle Cloud service, you must first create an instance of the service.
4. Create a project for visual Apps	To develop a visual application, you must create a VB Studio project based on the Visual Application template.
5. Set up the project for development	When you create a project, artifacts, created by default, require additional configuration before use.
a) Add the built-in free VM.	Build jobs in VB Studio run on Build VMs. You must manually add it.
b) Configure the deployment job	When you create a visual application project, the deployment job is missing the credentials to connect to the target development instance, so you must specify them manually.
c) Run the development build pipeline	Test the default package and build jobs to make sure they generate a build artifact and deploy it.
d) View the deployed visual application	After you've deployed the visual application, you can view it's URL on the Environments page.
6. Add users to the project	To allow your team members to access the visual application project, invite them to join the project.
7. Add the Visual Builder production instance to an environment	First, create an environment and add the Visual Builder production instance to it.
8. Create the packaging & deployment build jobs, and set up a build pipeline	So that your visual applications will be promoted to your Visual Builder production instance.



Steps for for Other Types of Applications

- If your team develops applications using popular build frameworks such as Maven, Gradle, Ant, and Node.js; or uses standard infrastructure interfaces such as Docker, Kubernetes, and Terraform, follow these steps to set up VB Studio:





Thank You