



Reactor

Java On Azure 动手实验营

Map



个人介绍



Kinfey Lo – (卢建晖)

Microsoft Cloud Advocate

前微软MVP、Xamarin MVP和微软RD，拥有超过10年的云原生、人工智能和移动应用经验，为教育、金融和医疗提供应用解决方案。 Microsoft Ignite, TechEd 会议讲师，Microsoft AI 黑客马拉松教练，目前在微软，为技术人员和不同行业宣讲技术和相关应用场景。



爱编程(Python , C# , TypeScript , Swift , Rust , Go)

专注于人工智能，云原生，跨平台移动开发

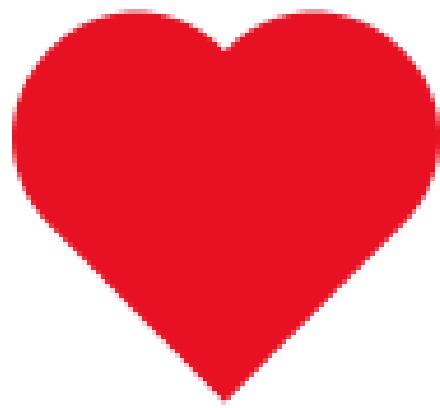
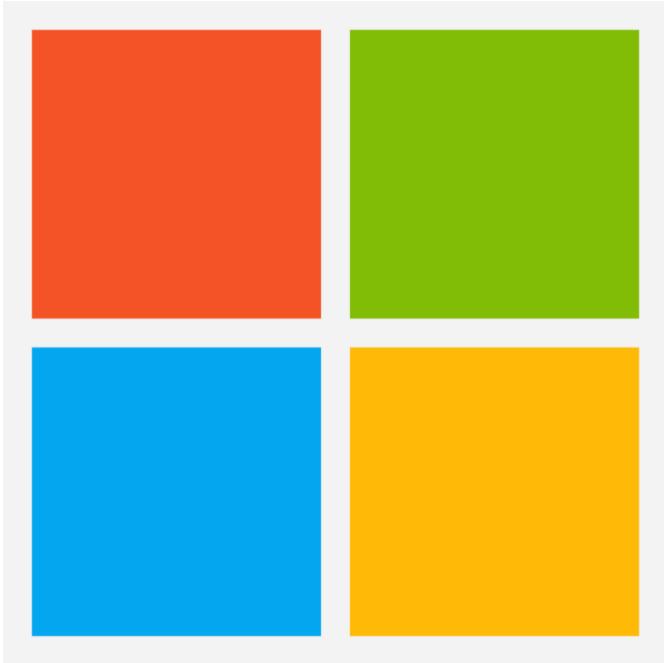
Github : <https://github.com/kinfey>

Email : kinfeylo@microsoft.com Blog : <https://blog.csdn.net/kinfey>

Twitter : @Ljh8304

Java,你好





微软在自有产品，技术都不少Java的身影



Microsoft JDK

<https://www.microsoft.com/openjdk>



Microsoft JDK

微软是OpenJDK的贡献者，更发布了Microsoft Open JDK。Microsoft 致力于与 OpenJDK 的行业合作伙伴合作，以实现 Java 生态系统的共同利益。我们的默认策略是在所有补丁被接受时将其上传。微软在2020年贡献WindowsOnARM 的Java集成，并为 macOS M1的Java集成的重要代码贡献。

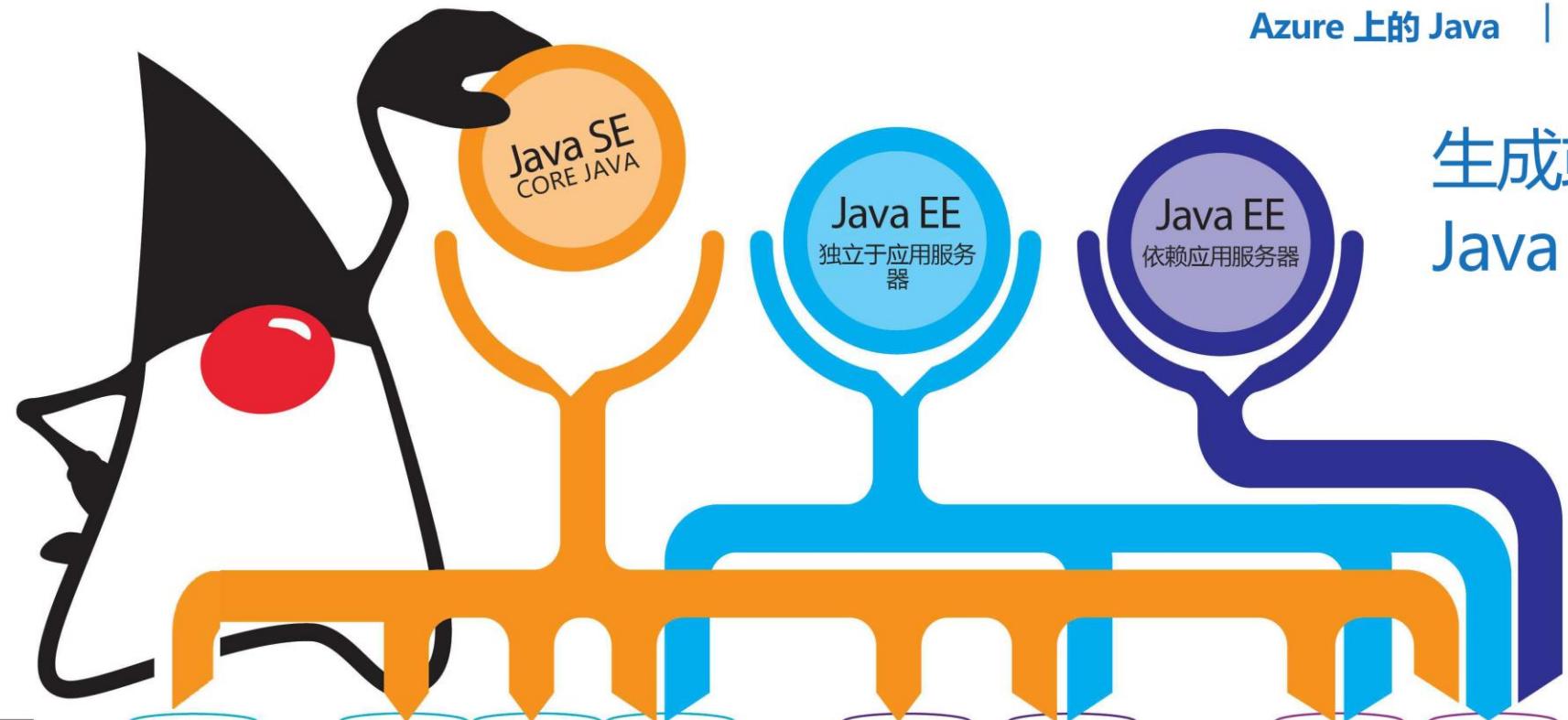
OpenJDK 11.0.12:

Platform	Architecture	Type	Download Link
Linux	x64	tar.gz	microsoft-jdk-11.0.12.7.1-linux-x64.tar.gz
macOS	x64	tar.gz	microsoft-jdk-11.0.12.7.1-macos-x64.tar.gz
macOS	x64	pkg	microsoft-jdk-11.0.12.7.1-macos-x64.pkg
Windows	x64	zip	microsoft-jdk-11.0.12.7.1-windows-x64.zip
Windows	x64	msi	microsoft-jdk-11.0.12.7.1-windows-x64.msi
Linux	AArch64 / ARM64	tar.gz	microsoft-jdk-11.0.12.7.1-linux-aarch64.tar.gz
Windows	AArch64 / ARM64	zip	microsoft-jdk-11.0.12.7.1-windows-aarch64.zip
Windows	AArch64 / ARM64	msi	microsoft-jdk-11.0.12.7.1-windows-aarch64.msi

OpenJDK 16.0.2:

Platform	Architecture	Type	Download Link
Linux	x64	tar.gz	microsoft-jdk-16.0.2.7.1-linux-x64.tar.gz
macOS	x64	tar.gz	microsoft-jdk-16.0.2.7.1-macos-x64.tar.gz
macOS	x64	pkg	microsoft-jdk-16.0.2.7.1-macos-x64.pkg
Windows	x64	zip	microsoft-jdk-16.0.2.7.1-windows-x64.zip
Windows	x64	msi	microsoft-jdk-16.0.2.7.1-windows-x64.msi
Linux	AArch64 / ARM64	tar.gz	microsoft-jdk-16.0.2.7.1-linux-aarch64.tar.gz
macOS	AArch64 / M1	tar.gz	microsoft-jdk-16.0.2.7.1-macos-aarch64.tar.gz
macOS	AArch64 / M1	pkg	microsoft-jdk-16.0.2.7.1-macos-aarch64.pkg
Windows	AArch64 / ARM64	zip	microsoft-jdk-16.0.2.7.1-windows-aarch64.zip
Windows	AArch64 / ARM64	msi	microsoft-jdk-16.0.2.7.1-windows-aarch64.msi

生成或迁移 Java 应用



Microsoft Azure

Visual Studio Code 调试 Java



或者 Java



Visual Studio Code For Java



Extension Pack for Java

v0.18.4 [Preview](#)

Microsoft | ⚡ 9,081,842 | ★★★★☆(39)

Popular extensions for Java development that provides Java IntelliSense, debugging, testing, Maven/Gradle support, project mana...

[Install](#) 

[Details](#) [Feature Contributions](#) [Changelog](#)

Extension Pack (6)



Debugger for Java
A lightweight Java debugger for Visual Studio...
Microsoft 



Language Support for Java(TM) by Red Hat
Java Linting, Intellisense, formatting, refactor...
Red Hat [Install](#)



Visual Studio IntelliCode
AI-assisted development
Microsoft [Install](#)



Maven for Java
Manage Maven projects, execute goals, gen...
Microsoft [Install](#)

Categories

[Programming Languages](#) [Snippets](#)
[Linters](#) [Debuggers](#) [Formatters](#)
[Extension Packs](#)

Resources

[Marketplace](#) [Repository](#) [License](#)

More Info

Visual Studio Code For Java



A screenshot of the Visual Studio Code interface, showing a Java application running. The top bar shows tabs for 'Welcome', 'App.java' (which is currently selected), 'launch.json', and 'Configure Java Runtime'. To the right of the tabs is a toolbar with various icons for file operations like copy, paste, and search. Below the top bar, the status bar displays 'src > App.java > ...' and includes 'Run | Debug' buttons. The main code editor area shows the following Java code:

```
1 public class App {  
2     Run | Debug  
3     public static void main(String[] args) throws Exception { args = String[0]@8  
4         System.out.println("Hello, World!");  
5     }  
6 }
```

The line 'System.out.println("Hello, World!");' is highlighted with a green background, indicating it is the current line of execution. A red circular icon with a white dot is visible on the far left edge of the code editor, likely indicating a break point or a specific marker.

Spring Boot 应用

- Spring-Boot是由Pivotal团队提供的全新框架，其设计目的是用来简化新Spring应用的初始搭建以及开发过程。个人理解来说Spring-Boot其实不是什么新的框架，它默认配置了很多框架的使用方式，就像maven整合了所有的jar包，Spring-Boot整合了其他相关联框架。
- Spring-Boot 基于Spring框架，而Spring-Cloud基于Spring-Boot，为微服务体系开发中的架构问题，提供了一整套的解决方案——服务注册与发现，服务消费，服务保护与熔断，网关，分布式调用追踪，分布式配置管理等。
- Spring-Boot 四大特性：自动配置、起步依赖、Actuator、命令行界面(CLI)



Visual Studio Code For Spring Boot



Spring Initializr Java Support v0.7.0

Microsoft | ↗ 1,080,584 | ★★★★☆(5)

A lightweight extension based on Spring Initializr to generate quick start Spring Boot Java projects.

Disable



Uninstall



This extension is enabled globally.

VARIABLES

Local

> name: "azure"

> this: APIController@55

WATCH

```

src > main > java > com > javahol > springbootdemo > APIController.java > APIController > getInfo(String)
1 package com.javahol.springbootdemo;
2
3 import org.springframework.web.bind.annotation.PathVariable;
4 import org.springframework.web.bind.annotation.RequestMapping;
5 import org.springframework.web.bind.annotation.RequestMethod;
6 import org.springframework.web.bind.annotation.RestController;
7
8 @RestController
9 @RequestMapping("api")
10 public class APIController {
11
12
13     @RequestMapping(value = "/getInfo/{name}",method = RequestMethod.GET)
14     public String getInfo(@PathVariable String name){ name = "azure"
15         return "Hi , " + name ;| name = "azure"
16     }
17
18 }
```

CALL STACK

ApplicationFilterChain.inter

ApplicationFilterChain.doFil

[Load All Stack Frames](#)

Thread [... PAUSED ON BREAKPOINT

APIController.getInfo(String)

NativeMethodAccessorImpl.in

NativeMethodAccessorImpl.in

DelegatingMethodAccessorImp

Method.invoke(Object, Object)

BREAKPOINTS

 Uncaught Exceptions Caught Exceptions APIController

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

zing Spring embedded WebApplicationContext
2021-08-30 23:11:00.783 INFO 21664 --- [restartedMain] w.s.c.ServletWebServerApplicationContext : Root Web
ApplicationContext: initialization completed in 1091 ms
2021-08-30 23:11:01.172 INFO 21664 --- [restartedMain] o.s.b.d.a.OptionalLiveReloadServer : LiveRelo
ad server is running on port 35729
2021-08-30 23:11:01.201 INFO 21664 --- [restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat s
tarted on port(s): 8080 (http) with context path ''
2021-08-30 23:11:01.213 INFO 21664 --- [restartedMain] c.j.s.SpringbootdemoApplication : Started
SpringbootdemoApplication in 1.957 seconds (JVM running for 2.498)
2021-08-30 23:11:09.106 INFO 21664 --- [nio-8080-exec-1] o.a.c.c.C.[Tomcat].[localhost].[/] : Initiali
zing Spring DispatcherServlet 'dispatcherServlet'
2021-08-30 23:11:09.108 INFO 21664 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet : Initiali
zing Servlet 'dispatcherServlet'
2021-08-30 23:11:09.109 INFO 21664 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet : Comple
te initialization in 1 ms

+

v

^

x

powershell

Java Debug ...

进入动手环节的第一步-基础环境+VSCode 调试Java程序



[00.AboutJava.pdf](#)

[00.Install.pdf](#)

[01.JavaWithVisualStudioCode.pdf](#)

[01.SpringbootWithVisualStudioCode.pdf](#)



1. 扫码关注
2. 进入公众号
3. 输入java0904 获取动手实验手册

Azure 部署 Java 应用



Azure 上的 Java



使用您喜欢的工具和框架进行开发

使用你最喜爱的 IDE (如 Eclipse、IntelliJ 和 Visual Studio Code for Java) 在 Azure 上生成、调试和部署 Java 应用程序。使用 Maven、Gradle 和 Jenkins 等生成和自动化工具，满足持续集成和持续交付需求。



通过完全托管服务加快交付速度

专注于构建业务应用程序，而不是管理您的基础设施。利用 Azure 应用服务、Azure Spring Cloud、Azure Kubernetes 服务 (AKS) 和 Azure Red Hat OpenShift 进行硬件和软件基础架构管理。



扩展您的 Java 应用程序

快速添加服务和功能，包括托管 MySQL、PostgreSQL 和 SQL 数据库、性能监控和机密管理。将您的应用程序与集成服务连接起来，例如具有 Java 消息服务 (JMS) 支持和 Azure API 管理的 Azure 服务总线。

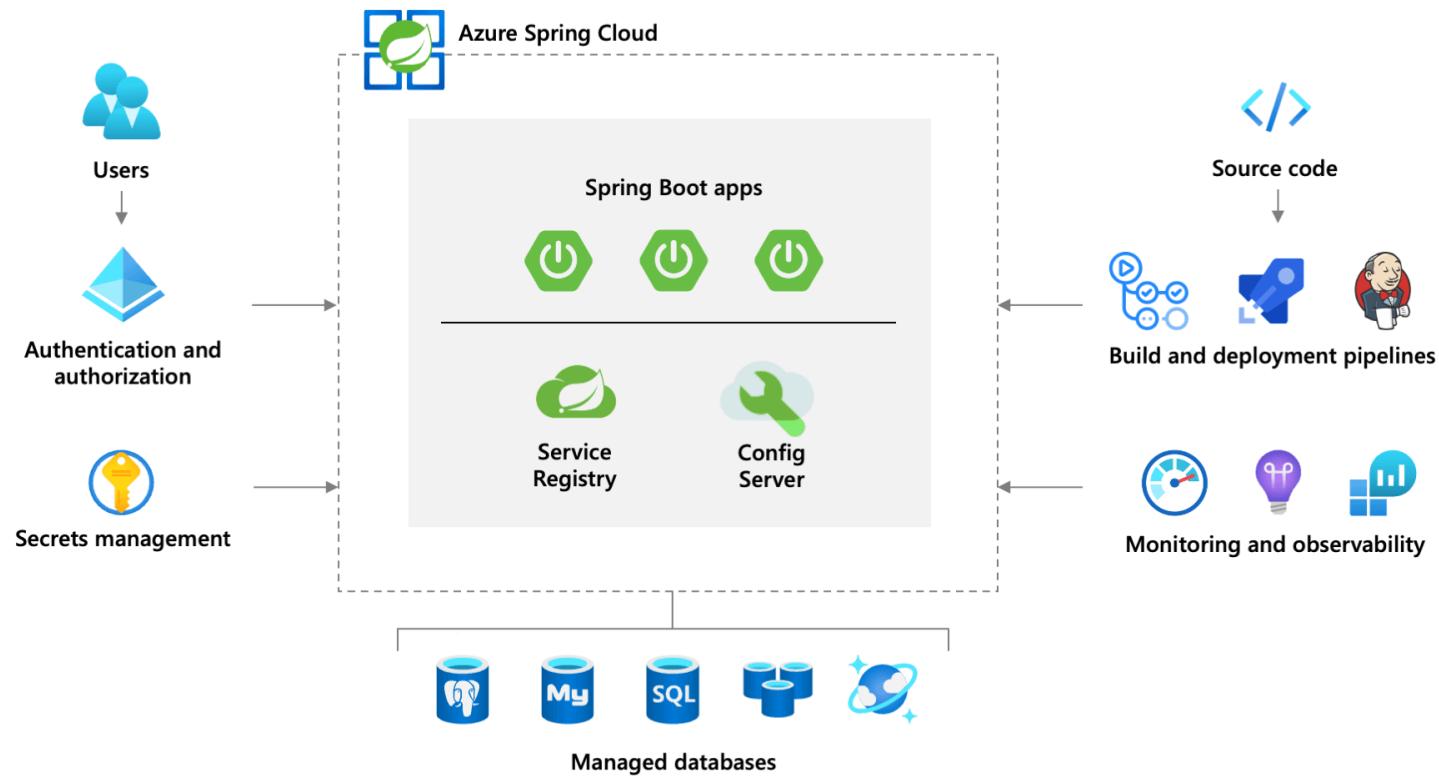


Microsoft 合作伙伴生态系统

获取一流的解决方案，将现有 Java 工作负载引入 Azure 并扩展应用程序的功能。访问越来越多的以 Java 为中心的解决方案组合，从具有联合开发和支持的独特托管选项到适用于流行 Linux 发行版的 Azure 市场映像。

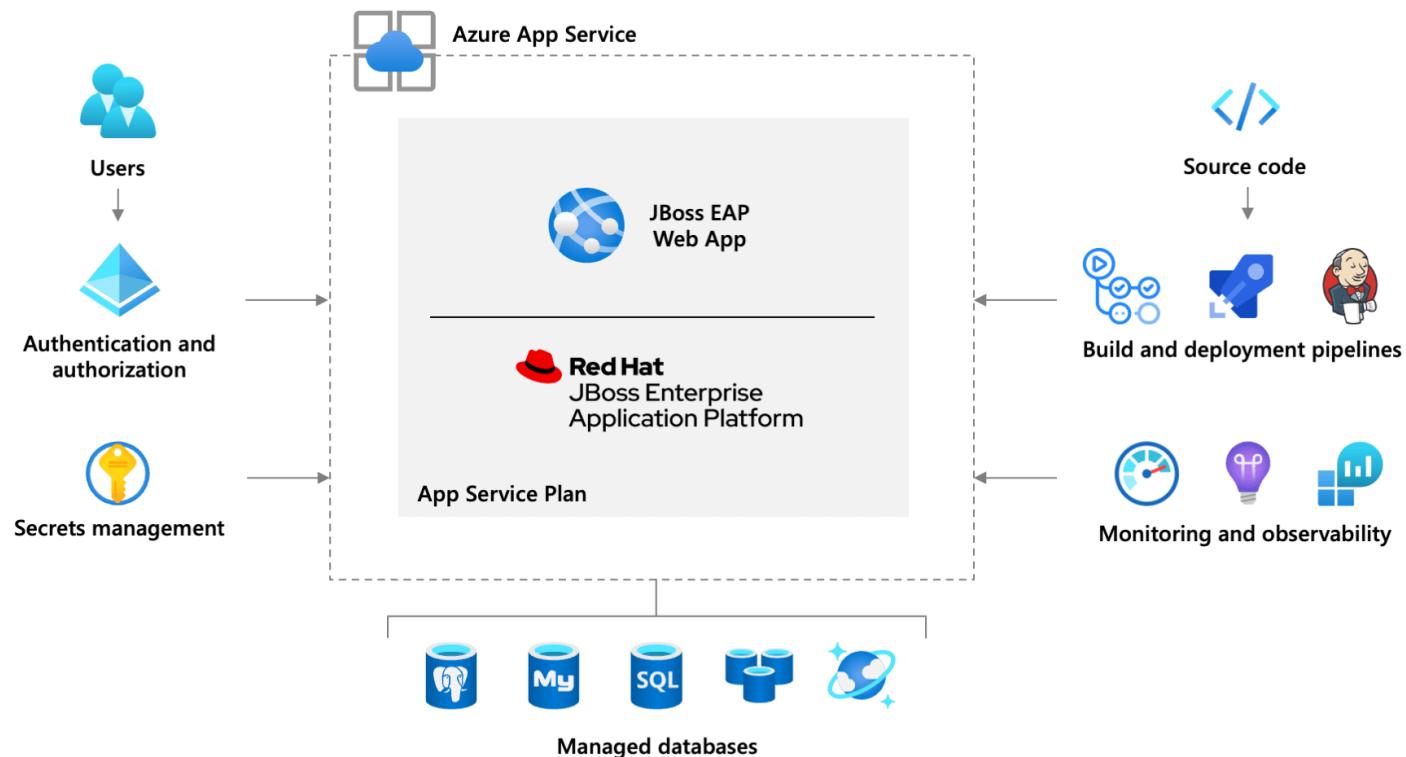
Azure上的Spring Cloud

使用 Azure Spring Cloud (能让你专注于构建运行业务的应用的一种服务) 管理 Spring Boot 应用程序。在完全托管的环境中轻松部署、运营和扩展应用。



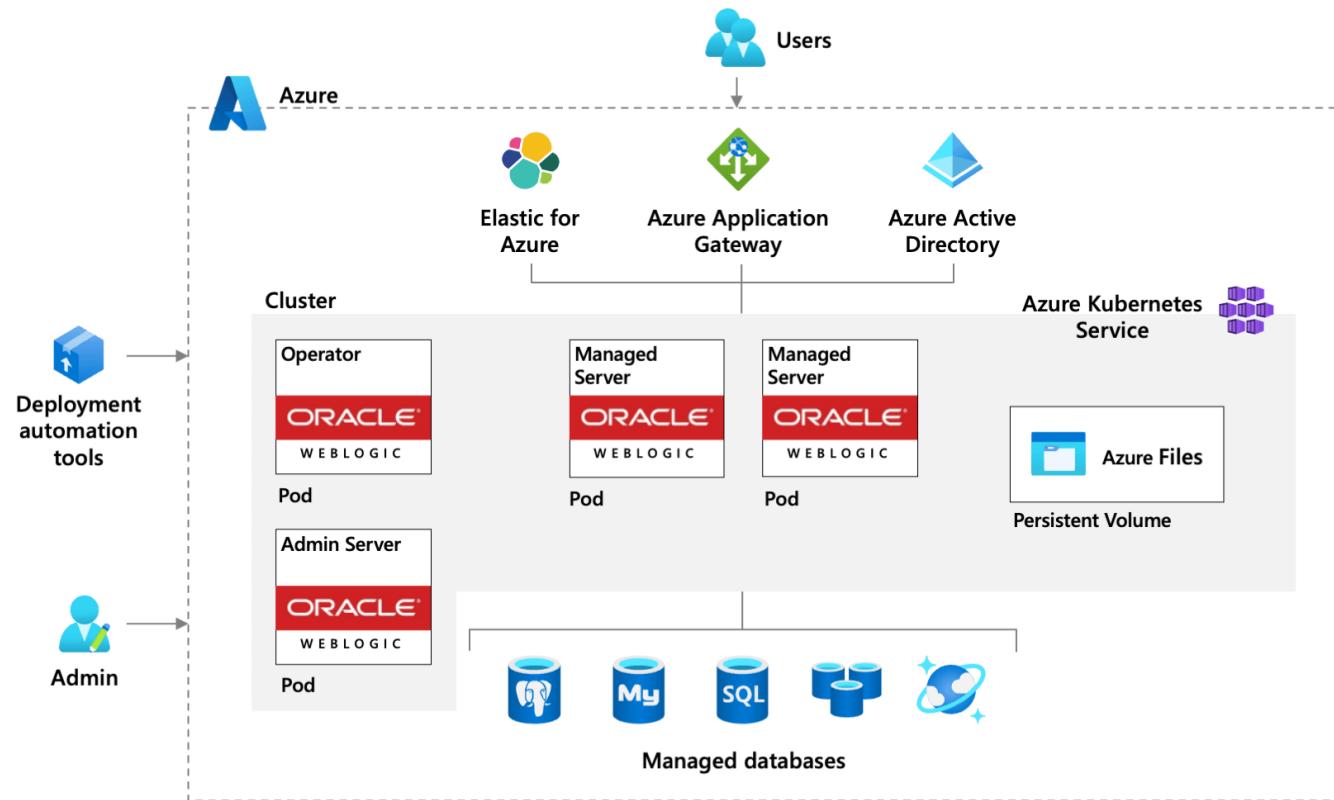
Azure上的JavaEE

在 Azure 应用服务上使用 Red Hat JBoss 企业应用程序平台 (EAP) 在完全托管的服务上部署企业 Java 应用。在 Microsoft 和 Red Hat 的共同支持下获得应用服务的完全托管体验。



Azure Kubernetes 服务 (AKS) 上的 Java EE

在 Azure 应用服务上使用 Red Hat JBoss 企业应用程序平台 (EAP) 在完全托管的服务上部署企业 Java 应用。在 Microsoft 和 Red Hat 的共同支持下获得应用服务的完全托管体验。



使用熟悉的服务在云中生成和部署 Java 应用

迁移协助



Azure 应用服务 Migration Assistant

使用应用服务迁移助手将 Tomcat 上运行的 Java 应用程序迁移到 Azure。

基于容器的现代化



Azure Migrate: 应用容器化

使用容器优化计算。开始使用自动化工具来容器化 Java Web 应用，并将这些应用移动到 Azure Kubernetes 服务 (AKS)。

托管的 POSTGRESQL、POSTGRESQL 和 SQL SERVER



Azure 数据库服务

使用 PostgreSQL、MySQL 和 SQL Server 的可靠、可缩放且完全托管的服务在云中设置你喜欢的关系数据库。

AI 和认知服务



Azure 认知服务

使用 AI 和认知 API 来生成智能应用。Azure 认知服务使每位开发人员无需具备机器学习的专业知识就能接触到 AI。

安全和标识服务



Azure Active Directory (Azure AD)

使用 Azure AD 向应用添加身份验证，然后在 Azure Key Vault 中存储加密密钥和其他机密。

监视服务



Azure Monitor

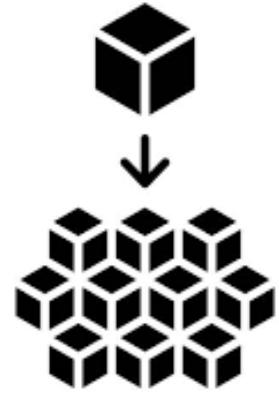
利用 Azure Monitor 收集、分析并处理来自 Azure 和本地环境的遥测数据，以最大限度地提升应用程序性能和可用性。

Azure 上的部署方式



整体式应用程序

单体式应用程序是一种一体化解决方案，它仍适用于许多项目。对于原型制作和初创公司，建议先从单体式应用程序开始。它的开发、调试、测试和部署过程都非常简单。传统上，单体式应用程序在应用程序服务器上运行，并将整个应用程序作为一个整体进行缩放。



微服务

当今企业需要通过适应和快速失败过程，对不确定因素进行预估，以取得成功。为了缩短上线时间，需要吸纳更多的工程师。但结果可能正相反！当更多的工程师需要协作时，需要耗费更多的精力进行沟通，从而减缓了上线速度。解决方法是分而治之。

在跨职能团队中，你可以尽可能独立地交付功能。每个团队都应对其软件的专用部分负责。



无服务器体系结构

函数只在事件发生时执行。函数由特定类型的事件“触发”。支持的触发器包括响应数据更改、响应消息、按计划运行或接收 HTTP 请求。



Batch作业

某些应用程序只需简单运行。它们执行特定的工作负载，然后退出，无需等待请求或用户输入。对于此用例，解决方案是使用批处理作业。尽管某些批处理可能持续数小时，但有时作业只需要运行一次或定期按计划间隔运行。

进入动手环节的第二步-部署Java应用到Azure



[02.PublishToAzure.pdf](#)

[02.SpringBootMicroServiceToAzure.pdf](#)



1. 扫码关注
2. 进入公众号
3. 输入java0904 获取动手实验手册

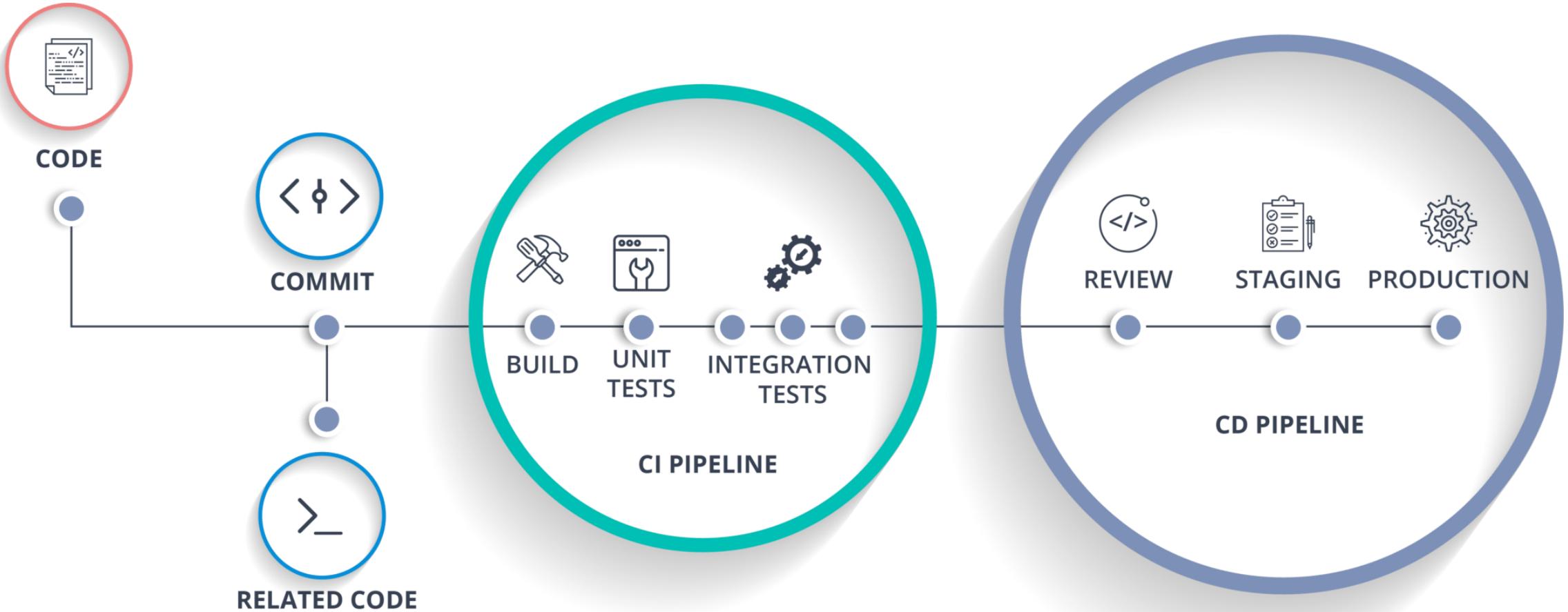
通过GitHub Action完成CI/CD



Microsoft & Github



CI/CD



GitHub Action 介绍

在 GitHub Actions 的仓库中自动化、自定义和执行软件开发工作流程。您可以发现、创建和共享操作以执行您喜欢的任何作业（包括 CI/CD），并将操作合并到完全自定义的工作流程中。

<https://github.com/features/actions>



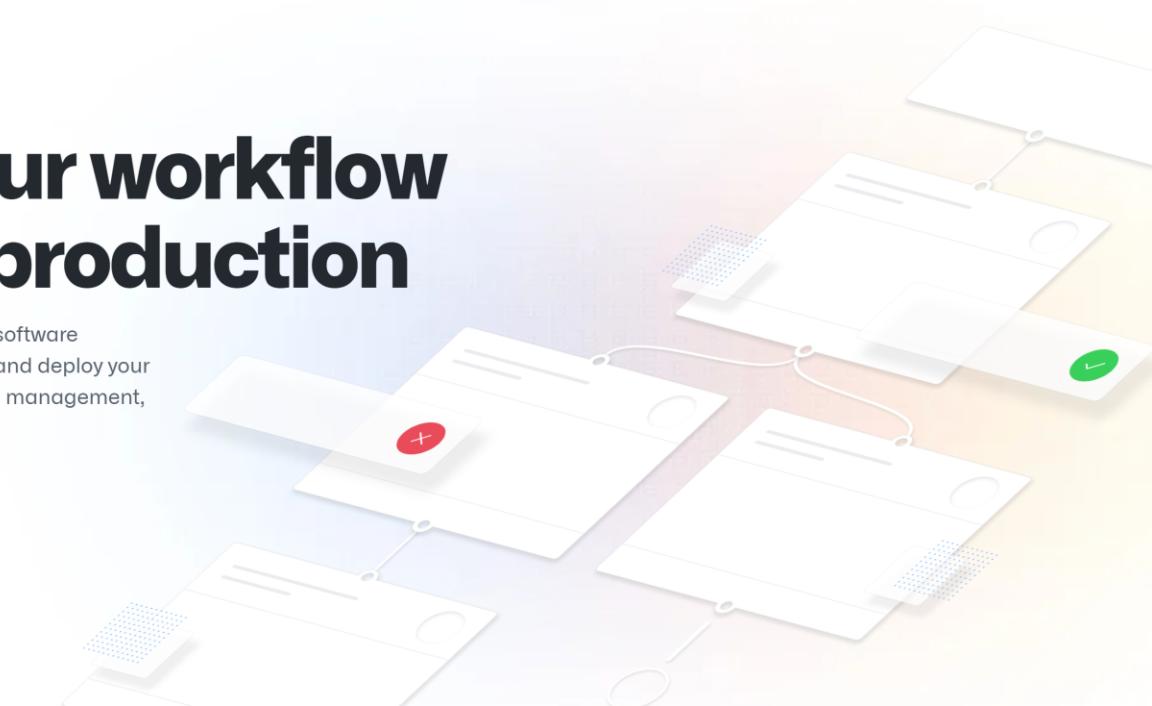
GitHub Actions

Automate your workflow from idea to production

GitHub Actions makes it easy to automate all your software workflows, now with world-class CI/CD. Build, test, and deploy your code right from GitHub. Make code reviews, branch management, and issue triaging work the way you want.

[Get started with Actions](#)

Questions? [Contact Sales →](#)



GitHub Action for Azure

<https://azure.github.io/actions/>

自动化工作流程

在任何 GitHub 事件（例如代码提交、创建拉取请求或新的 GitHub 版本）上轻松自动化代码到云的工作流程。跨流行的语言和框架（包括 .NET、Node.js、Java、PHP、Ruby、Python）在容器中或在任何操作系统上部署应用程序到 Azure。

安全性和合规性

使用 GitHub Actions for Azure 将治理、安全性和合规性自动化到软件开发生命周期的早期阶段。编写基础设施配置、发布管道、合规性和安全策略“即代码”，以实现持续改进和更高的透明度。

无缝运行

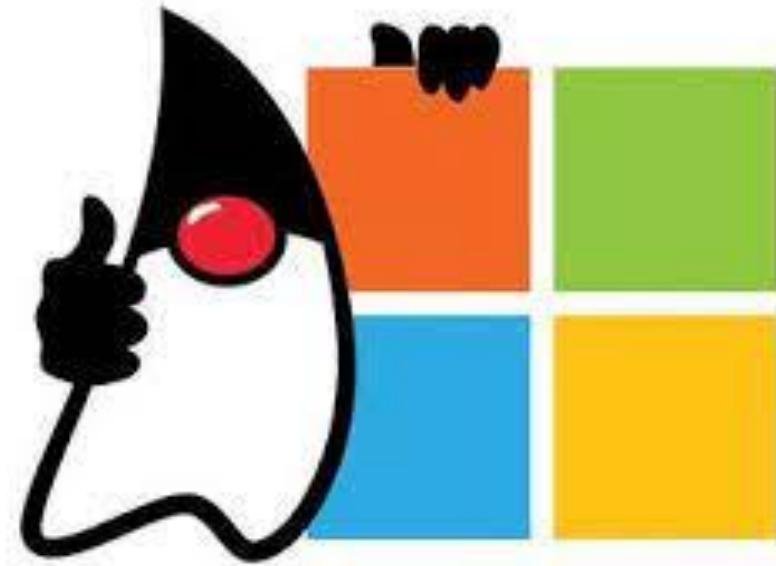
从 IT 环境的手动管理和操作转变为在企业范围内实现可控、协作和可重复的无缝自动化 IT 管理 (GitOps)。

加快开发者开发速度

使用与 GitHub Actions 集成的流行开发人员工具部署到 Azure。通过 Visual Studio Code、Azure CLI 和 Azure 门户的内置扩展，显着减少启动时间，避免频繁的上下文切换，并帮助团队提高工作效率。

进入动手环节的第三步-用Github Action部署Java应用到Azure

03.DeployToGitHubAction.pdf



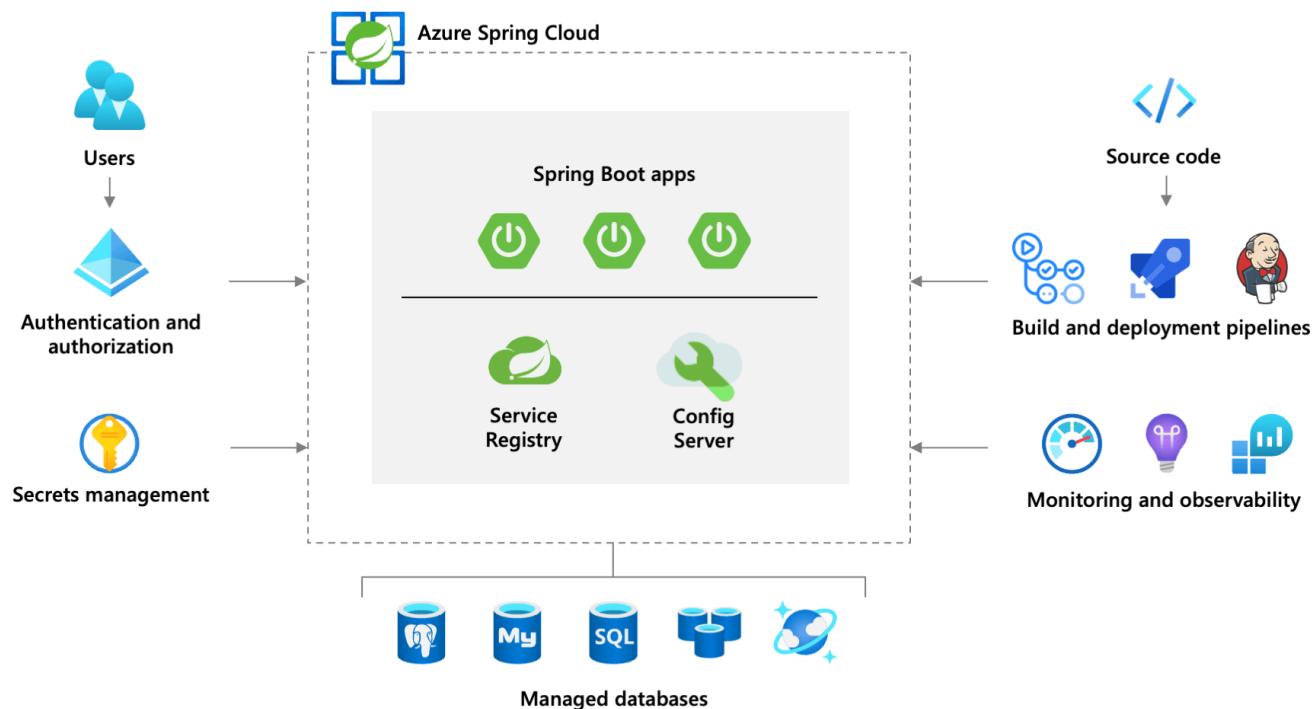
1. 扫码关注
2. 进入公众号
3. 输入java0904 获取动手实验手册

Spring Cloud



Azure Spring Cloud

借助 Azure Spring Cloud，可以轻松地将 Spring Boot 微服务应用程序部署到 Azure，不需更改任何代码。该服务管理 Spring Cloud 应用程序的基础结构，因此开发人员可以专注于其代码。Azure Spring Cloud 可以通过以下方法提供生命周期管理：综合性监视和诊断、配置管理、服务发现、CI/CD 集成、蓝绿部署等。



Azure Spring Cloud

将应用程序部署到 Azure Spring Cloud 有很多好处。方法

:

有效地迁移现有 Spring 应用并管理云缩放和成本。使用 Spring Cloud 模式实现应用现代化，以提高敏捷性和交付速度。以云规模运行 Java，并在不使用复杂基础结构的情况下提高利用率。无需依赖容器化即可快速开发和部署。有效轻松地监视生产工作负载。Azure Spring Cloud 同时支持 Java Spring Boot 和 ASP.NET Core Steeltoe 应用。Steeltoe 支持目前以公共预览版的形式提供。使用公共预览版产品/服务，你可以在产品/服务正式发布之前体验新功能。

Azure Spring Cloud 是 Azure 生态系统的一部分，用户可以通过它轻松绑定到其他 Azure 服务，其中包括存储、数据库、监视等。

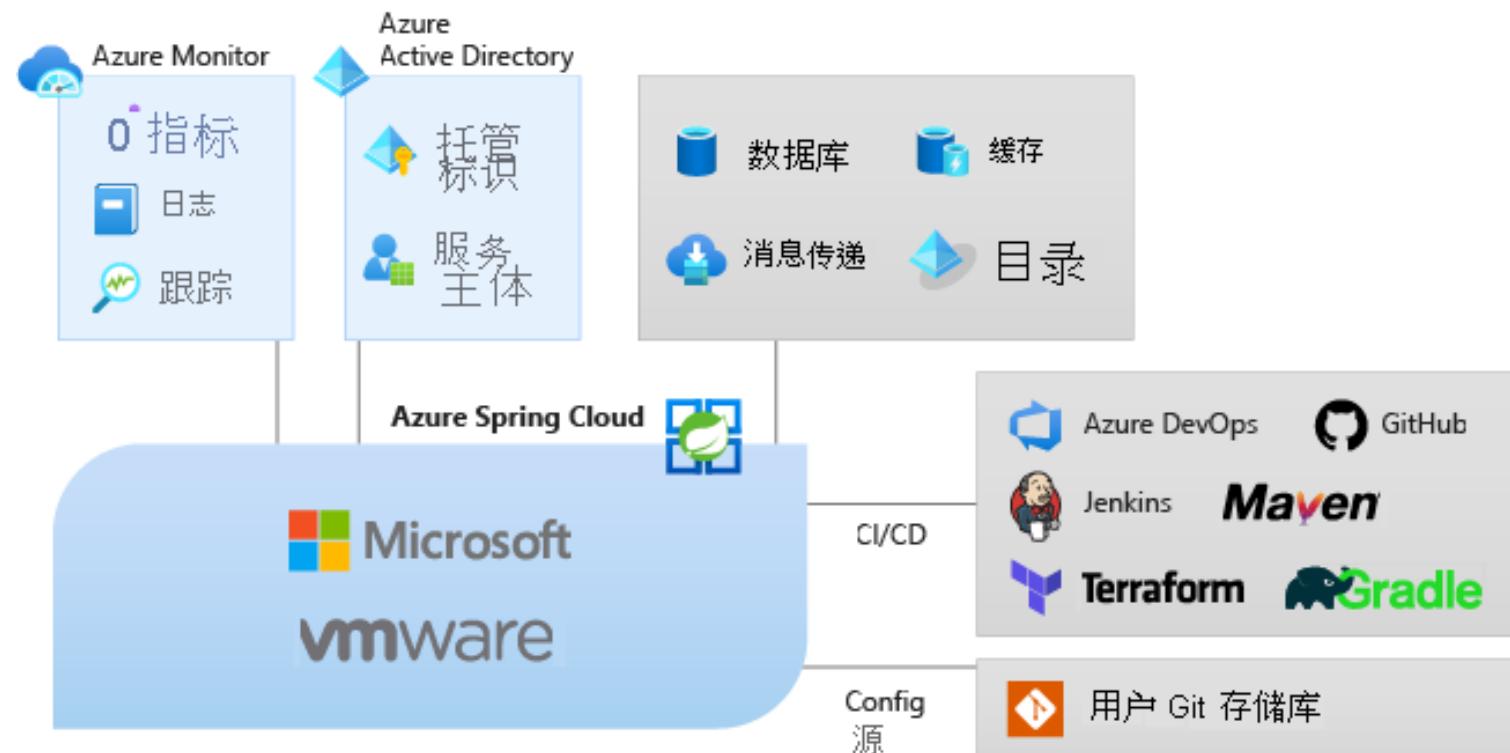
Azure Spring Cloud 是适用于 Spring Boot 应用的完全托管服务，可让你专注于构建和运行应用，而无需管理基础设施。

只需为 Spring Boot 应用部署 JAR 或代码，或为 Steeltoe 应用部署 Zip，Azure Spring Cloud 就会自动将应用与 Spring 服务运行时和内置的应用生命周期关联起来。

监视也很简单。部署后，可以监视应用性能，修复错误以及快速改进应用程序。

与 Azure 的生态系统和服务的完全集成。

Azure Spring Cloud 具有完全托管的基础结构和内置的生命周期管理且易于监视，可供企业使用。



CosmosDB

Azure Cosmos DB 是一种用于新式应用开发的完全托管的 NoSQL 数据库。个位数的毫秒响应时间以及自动和即时的可伸缩性，可保证任何规模的速度。SLA 支持的可用性和企业级安全性可确保业务连续性。得益于全球各地的统一多区域数据分布、适用于常用语言的开源 API 和 SDK，应用的开发速度更快、效率更高。作为一项完全托管的服务，Azure Cosmos DB 使用自动管理、更新和修补，使你无需进行数据库管理。它还通过经济高效的无服务器和自动缩放选项处理容量管理，这些选项可响应应用程序的需求，使容量与需求相匹配。



在任何规模下都能保证速度（即使遇到突发）：在世界任何地方都能实现即时无限弹性、快速读取和多主数据库写入



使用适用于热门语言的 SDK、本机 Core (SQL) API 以及适用于 MongoDB、Cassandra、Gremlin 和非 ETL（提取、转换、加载）分析功能的 API 进行快速、灵活的应用开发



适用于任务关键应用程序：具有有保证的业务连续性、99.999% 的可用性和企业级安全性



完全托管且经济高效的无服务器数据库：具有即时、自动缩放功能（可以响应应用程序需求）、集成的缓存和基于使用量的定价选项

CosmosDB

- 与新式（云原生）应用开发中使用的关键 Azure 服务深度集成，包括 Azure Functions、IoT 中心、AKS (Azure Kubernetes Service)、应用服务等。
- 从多个数据库 API 中进行选择，包括本机 Core (SQL) API、适用于 MongoDB 的 API、Cassandra API、Gremlin API 和表 API。
- 使用所选语言和适用于 .NET、Java、Node.js 和 Python 的 SDK，在 Core (SQL) API 上构建应用。或者为任何其他数据库 API 选择的驱动程序。
- 使用 Azure Synapse Analytics 对 Azure Cosmos DB 中存储的近乎实时的操作数据运行无 ETL 分析。
- 通过更改源，可以轻松地跟踪和管理对数据库容器所做的更改以及使用 Azure Functions 创建触发事件。
- Azure Cosmos DB 的无架构服务会自动为所有数据（无论数据模型为何）编制索引，从而快速交付查询。



进入动手环节的第四步-部署Java应用到Azure Spring Cloud

04.AzureSpringCloudWithCosmosDB.pdf

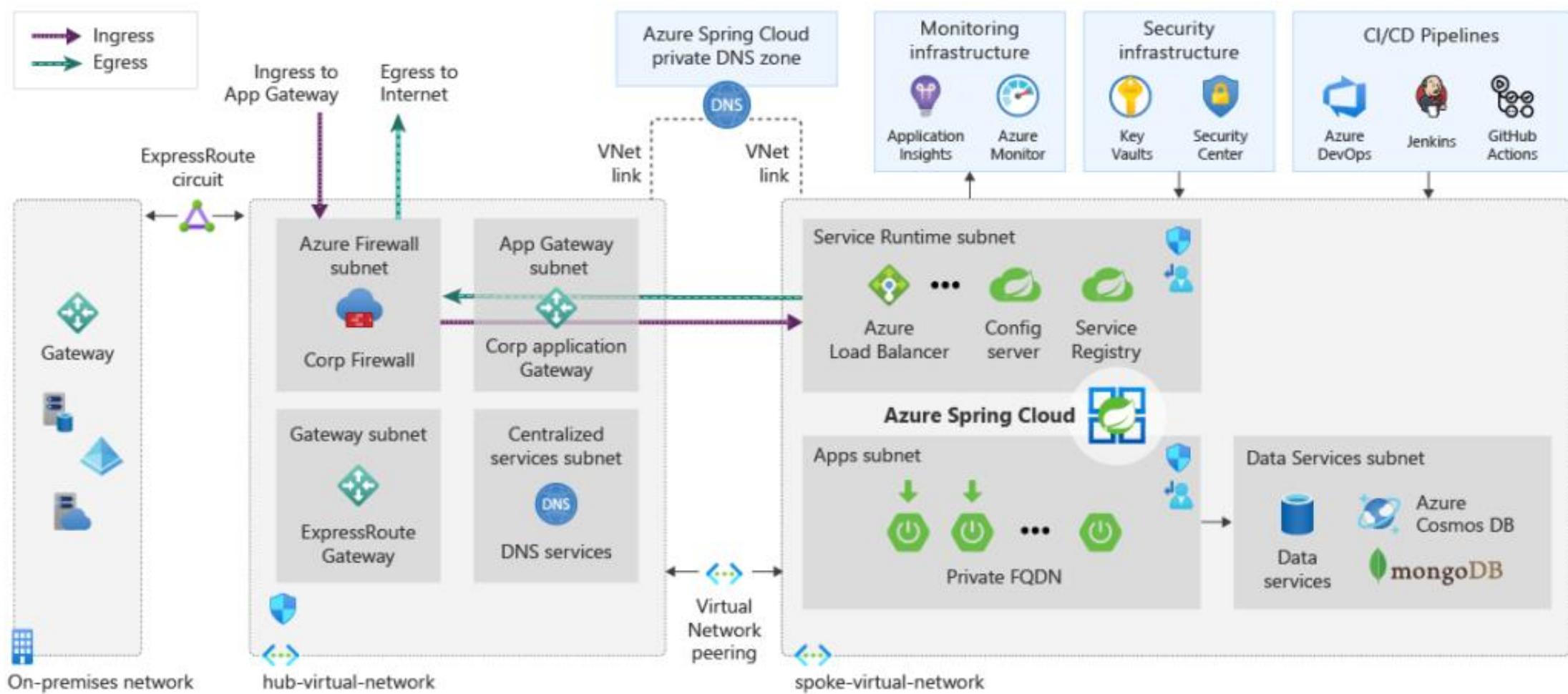


1. 扫码关注
2. 进入公众号
3. 输入java0904 获取动手实验手册

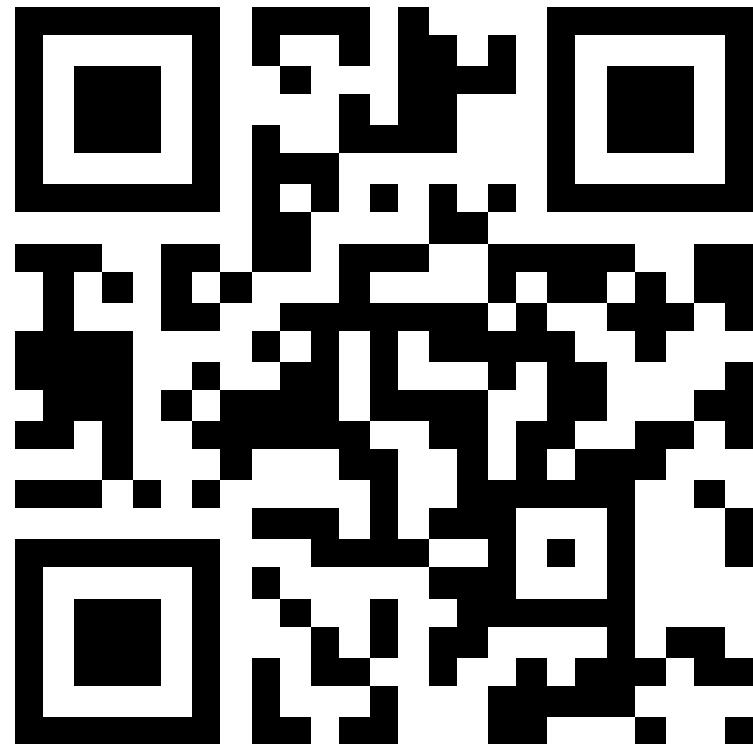
小结



Java Azure Solution



MS Learn 的学习模块推荐



Azure 上的 Java

<https://aka.ms/JavaAzureHOL>



Reactor

Thank You!