



第二版：SpringCloud 29 道

目录

第二版：SpringCloud 29 道	1
一、什么是 Spring Cloud?	2
二、什么是 Spring Boot?	2
三、如何覆盖 Spring Boot 项目的默认属性?	2
四、Actuator 在 Spring Boot 中的作用	2
五、如何在 Spring Boot 应用程序中实现 Spring 安全性?	3
六、Spring Boot 支持哪些嵌入式容器?	4
七、微服务的端到端测试意味着什么?	4
八、什么是 Semantic 监控?	4
九如何设置服务发现?	4
十、为什么要选择微服务架构?	5
十一、为什么在微服务中需要 Reports 报告和 Dashboards 仪表板?	5
十二、为什么人们会犹豫使用微服务?	5
十三、PACT 如何运作?	6
十四、谈一下领域驱动设计	6
十五、什么是耦合和凝聚力?	6
十六、什么是 Oauth?	7
十七、为什么我们需要微服务容器?	7
十八、访问 RESTful 微服务的方法是什么?	7
十九、微服务测试的主要障碍是什么?	7
二十、过渡到微服务时的常见错误	8
二十一、微服务设计的基础是什么?	8
二十二、在 Spring MVC 应用程序中使用 WebMvcTest 注释有什么用处?	9
二十三、什么是有界上下文?	9
二十四、什么是不同类型的双因素身份认证?	9
二十五、什么是客户证书?	9
二十六、康威定律是什么?	10
二十七、如何配置 Spring Boot 应用程序日志记录?	10

微信搜一搜

搜云库技术团队



二十八、您将如何在微服务上执行安全测试?	10
二十九、什么是幂等性?它是如何使用的?	10

我们的网站: <https://tech.souyunku.com>

关注我们的公众号: **搜云库技术团队**, **回复以下关键字**

回复: **【进群】** 邀请您进「**技术架构分享群**」

回复: **【内推】** 即可进: 北京, 上海, 广州, 深圳, 杭州, 成都, 武汉, 南京, 郑州, 西安, 长沙「**程序员工作内推群**」

回复 **【1024】** 送 4000G 最新架构师视频

回复 **【PPT】** 即可无套路获取, 以下最新整理调优 PPT!

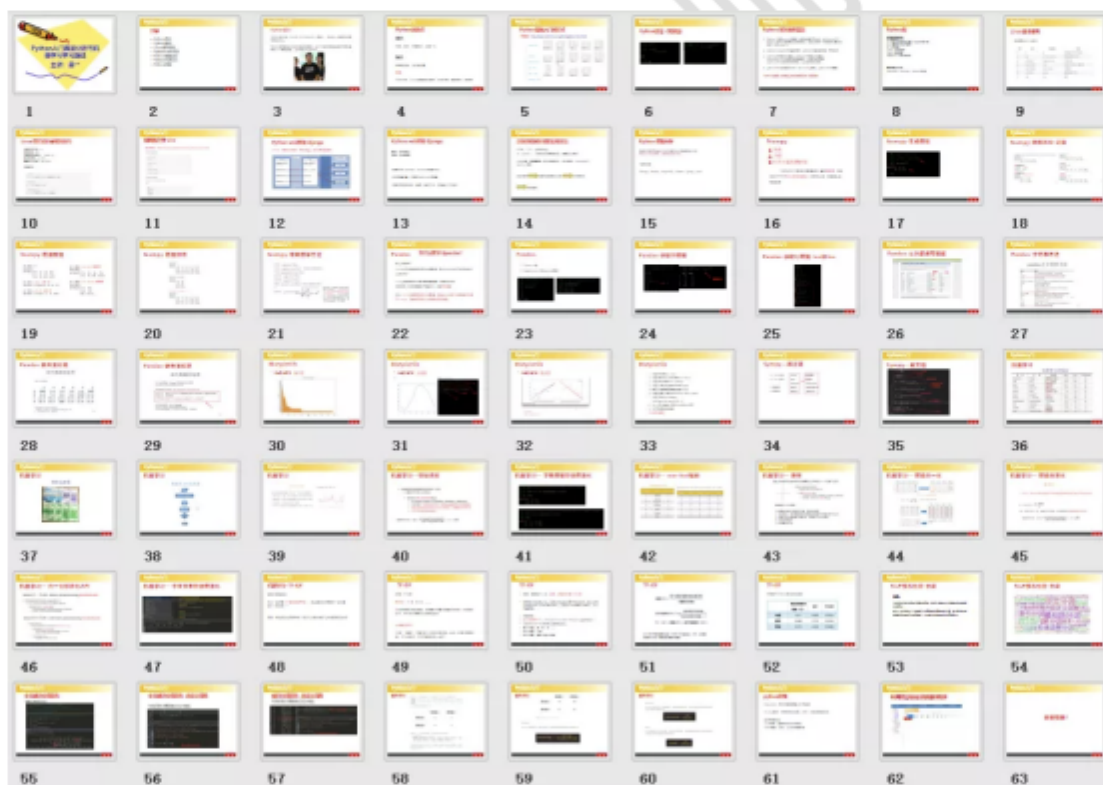
46 页《JVM 深度调优, 演讲 PPT》



53 页《Elasticsearch 调优演讲 PPT》



63 页《Python 数据分析入门 PPT》





微信扫一扫

<https://tech.souyunku.com>

技术、架构、资料、工作、内推
专注于分享最有价值的互联网技术干货文章

一、什么是 Spring Cloud?

在微服务中，SpringCloud 是一个提供与外部系统集成的系统。它是一个敏捷的框架，可以短平快构建应用程序。与有限数量的数据处理相关联，它在微服务体系结构中起着非常重要的作用。 以下为 **Spring Cloud** 的核心特性：

- 1、 版本化/分布式配置。
- 2、 服务注册和发现。
- 3、 服务和服务之间的调用。
- 4、 路由。
- 5、 断路器和负载平衡。
- 6、 分布式消息传递。

二、什么是 Spring Boot?

Spring boot 是微服务面试问题的主要话题。 随着新功能的加入，Spring 变得越来越复杂。无论何时启动新项目，都必须添加新的构建路径或 Maven 依赖项。简而言之，你需要从头开始做每件事。Spring Boot 是一种帮助您避免所有代码配置的解决方案。



三、如何覆盖 Spring Boot 项目的默认属性?

这可以通过在 application.properties 文件中指定属性来完成。例如, 在 Spring MVC 应用程序中, 您必须指定后缀和前缀。这可以通过在 application.properties 文件中输入下面提到的属性来完成。

- 对于后缀 - spring.mvc.view.suffix: .jsp
- 对于前缀 - spring.mvc.view.prefix: /WEB-INF/

四、Actuator 在 Spring Boot 中的作用

它是最重要的功能之一, 可帮助您访问在生产环境中运行的应用程序的当前状态。有多个指标可用于检查当前状态。它们还为 RESTful Web 服务提供端点, 可以简单地用于检查不同的度量标准。

五、如何在 Spring Boot 应用程序中实现 Spring 安全性?

实施需要最少的配置。您需要做的就是 spring-boot-starter-security 在 pom.xml 文件中添加 starter。您还需要创建一个 Spring 配置类, 它将覆盖所需的方法, 同时扩展 WebSecurityConfigurerAdapter 应用程序中的安全性。这是一些示例代码:

```
package com.gkatzioura.security.securityendpoints.config;
import org.springframework.context.annotation.Configuration;
```



```
import
org.springframework.security.config.annotation.web.builders.HttpSecurity;
import
org.springframework.security.config.annotation.web.configuration.Web
SecurityConfigurerAdapter;
@Configuration
public class SecurityConfig extends WebSecurityConfigurerAdapter {
@Override
protected void configure(HttpSecurity http) throws Exception {
http.authorizeRequests()
.antMatchers("/welcome").permitAll()
.anyRequest().authenticated()
.and()
.formLogin()
.permitAll()
.and()
.logout()
.permitAll();
}
}
```

六、Spring Boot 支持哪些嵌入式容器？

无论何时创建 Java 应用程序，都可以通过两种方法进行部署：使用外部的应用程序容器。将容器嵌入 jar 文件中。Spring Boot 包含 Jetty, Tomcat 和 Undertow 服务器，所有服务器都是嵌入式的。Jetty - 用于大量项目，Eclipse Jetty 可以嵌入到框架，应用程序服务器，工具和集群中。Tomcat - Apache Tomcat 是一个开源 JavaServer Pages 实现，可以很好地与嵌入式系统配合使用。



Undertow - 一个灵活而突出的 Web 服务器，它使用小型单一处理程序来开发 Web 服务器。

七、微服务的端到端测试意味着什么？

端到端测试 验证工作流中的所有流程，以检查一切是否按预期工作。它还确保系统以统一的方式工作，从而满足业务需求。

八、什么是 Semantic 监控？

它结合了对整个应用程序的监控以及自动化测试。语义监控的主要好处是找出对您的业务更有利可图的因素。从业务角度来看，语义监控以及服务层监控可以监控微服务。一旦检测到问题，它们就可以实现更快的隔离和 错误分类，从而减少修复所需的主要时间。它对服务层和事务层进行分类，以确定受可用性或性能不佳影响的事务。

九如何设置服务发现？

有多种方法可以设置服务发现。我将选择我认为效率最高的那个，Netflix 的 Eureka。这是一个简单的程序，不会对应用程序造成太大影响。此外，它支持多种类型的 Web 应用程序。Eureka 配置包括两个步骤 - 客户端配置和服务端配置。

使用属性文件可以轻松完成客户端配置。在 classpath 中，Eureka 搜索一个 eureka-client.properties 文件。它还搜索由特定于环境的属性文件中的环境引起的覆盖。



对于服务器配置，您必须首先配置客户端。完成后，服务器启动一个客户端，该客户端用于查找其他服务器。。默认情况下，Eureka 服务器使用客户端配置来查找对等服务器。

十、为什么要选择微服务架构？

这是一个非常常见的微服务面试问题，你应该准备好了！微服务架构提供了许多优点。这里有几个：

- 1、 微服务可以轻松适应其他框架或技术。
- 2、 单个进程的失败不会影响整个系统。
- 3、 为大企业和小型团队提供支持。
- 4、 可以在相对较短的时间内独立部署。

十一、为什么在微服务中需要 Reports 报告和 Dashboards 仪表板？

报告和仪表板主要用于监视和维护微服务。有多种工具可以帮助实现此目的。报告和仪表板可用于：找出哪些微服务公开了哪些资源。找出组件发生变化时受影响的服务。提供一个简单的点，只需要文档，就可以访问它。部署的组件的版本。

十二、为什么人们会犹豫使用微服务？

我见过许多开发者在这个问题上摸索。毕竟，在面试微服务架构师角色时，他们会被问到这个问题，所以承认它的缺点可能有点棘手。以下是一些很好的答案：



它们需要大量协作 - 微服务需要大量的合作。不同的微服务模块，可能分散在不同的团队，团队之间需要始终保持良好的同步。

他们需要建立繁重的架构 - 系统是分布式的，架构涉及很多。他们需要过多的计划来处理操作开销 - 如果您计划使用微服务架构，则需要为操作开销做好准备。需要熟练的专业人员，他们可以支持异构分布的微服务。

十三、PACT 如何运作？

PACT 是一个开源工具。它有助于测试消费者和服务提供商之间的互动。消费者服务开发人员首先编写一个测试，该测试定义了与服务提供者的交互模式。测试包括提供者的状态，请求正文和预期的响应。基于此，PACT 创建了一个针对其执行测试的存根。输出存储在 JSON 文件中。

十四、谈一下领域驱动设计

主要关注核心领域逻辑。基于领域的模型检测复杂设计。这涉及与公司层面领域方面的专家定期合作，以解决与领域相关的问题并改进应用程序的模型。在回答这个微服务面试问题时，您还需要提及 DDD 的核心基础知识。他们是：

- DDD 主要关注领域逻辑和领域本身。
- 复杂的设计完全基于领域的模型。
- 为了改进模型的设计并解决任何新出现的问题，DDD 不断与公司领域方面的专家合作。

十五、什么是耦合和凝聚力？



组件之间依赖关系强度的度量被认为是耦合。一个好的设计总是被认为具有高内聚力和低耦合性。面试官经常会问起凝聚力。它也是另一个测量单位。更像是一个模块内部的元素保持结合的程度。必须记住，设计微服务的一个重要关键是低耦合和高内聚的组合。当低耦合时，服务对其他服务的依赖很少。这样可以保持服务的完整性。在高内聚性中，将所有相关逻辑保存在服务中成为可能。否则，服务将尝试彼此通信，从而影响整体性能。

十六、什么是 Oauth?

开放授权协议，这允许通过在 HTTP 服务上启用客户端应用程序（例如第三方提供商 Facebook，GitHub 等）来访问资源所有者的资源。因此，您可以在不使用其凭据的情况下与另一个站点共享存储在一个站点上的资源。

OAuth 允许像 Facebook 这样的第三方使用最终用户的帐户信息，同时保证其安全（不使用或暴露用户的密码）。它更像是代表用户的中介，同时为服务器提供访问所需信息的令牌。

十七、为什么我们需要微服务容器?

要管理基于微服务的应用程序，容器是最简单的选择。它帮助用户单独部署和开发。您还可以使用 Docker 将微服务封装到容器的镜像中。没有任何额外的依赖或工作，微服务可以使用这些元素。

十八、访问 RESTful 微服务的方法是什么?

另一个经常被问到的微服务面试问题是如何访问 RESTful 微服务？你可以通过两种方法做到这一点：



- 使用负载均衡的 REST 模板。
- 使用多个微服务。

十九、微服务测试的主要障碍是什么？

说到缺点，这里是另一个微服务面试问题，将围绕测试微服务时面临的挑战。

- 在开始编写集成测试的测试用例之前，测试人员应该全面了解对所有入站和出站过程。
- 当独立的团队正在开发不同的功能时，协作可能会被证明是一项非常困难的任务。很难找到空闲时间窗口来执行完整的回归测试。
- 随着微服务数量的增加，系统的复杂性也随之增加。
- 在从单片架构过渡期间，测试人员必须确保组件之间的内部通信没有中断。

二十、过渡到微服务时的常见错误

不仅在开发上，而且在方面流程也经常发生错误。一些常见错误是：

- 通常开发人员无法概述当前的挑战。
- 重写已经存在的程序。
- 职责、时间线和界限没有明确定义。
- 未能从一开始就实施和确定自动化的范围。

二十一、微服务设计的基础是什么？



这可能是最常见的微服务面试问题之一。在回答这个问题时，你需要记住以下内容：

- 1、 定义范围。
- 2、 结合低耦合和高内聚。
- 3、 创建一个有唯一标识的服务，唯一标识将充当识别源，非常像数据库表中的唯一键。
- 4、 创建正确的 API 并在集成过程中特别注意。
- 5、 限制对数据的访问并将其限制到所需级别。
- 6、 在请求和响应之间保持顺畅的流程。
- 7、 自动化大多数流程，以减少时间复杂性。
- 8、 将表的数量保持在最低水平，以减少空间复杂性。
- 9、 不断监控架构，发现缺陷及时修复。
- 10、 每个微服务的数据存储应该分开。
- 11、 对于每个微服务，都应该有一个独立的构建。
- 12、 将微服务部署到容器中。
- 13、 服务器应被视为无状态。

二十二、在 Spring MVC 应用程序中使用 WebMvcTest 注释有什么用处？

WebMvcTest 注释用于单元测试 Spring MVC 应用程序。我们只想启动 ToTestController。执行此单元测试时，不会启动所有其他控制器和映射。

```
@WebMvcTest(value = ToTestController.class, secure = false);
```

二十三、什么是有界上下文？



有界上下文是域驱动设计的核心模式。DDD 战略设计部门的重点是处理大型模型和团队。DDD 通过将大型模型划分为不同的有界上下文并明确其相互关系来处理大型模型。

二十四、什么是不同类型的双因素身份认证？

执行双因素身份验证需要三种类型的凭据：

- 一件你知道的事情——比如密码、密码或屏幕锁定模式。
- 您拥有的物理凭证，如 OTP、电话或 ATM 卡，换句话说，您在外部或第三方设备中拥有的任何类型的凭证。
- 您的物理身份—如语音认证或生物特征安全，如指纹或眼睛扫描仪。

二十五、什么是客户证书？

客户端系统用于向远程服务器发出经过身份验证的请求的一种数字证书称为客户端证书。客户端证书在许多相互认证设计中起着非常重要的作用，为请求者的身份提供了强有力的保证。

二十六、康威定律是什么？

康威定律指出，“设计系统的组织，其产生的设计等同于组织之内、组织之间的沟通结构。” 面试官可能会问反微服务面试问题，比如康威定律与微服务的关系。一些松散耦合的 api 形成了微服务的体系结构。这种结构非常适合小团队实现自治组件的方式。这种体系结构使组织在重组其工作流程时更加灵活。



二十七、如何配置 Spring Boot 应用程序日志记录？

Spring Boot 附带了对 Log4J2, Java Util Logging 和 Logback 的支持。它通常预先配置为控制台输出。可以通过仅在 application.properties 文件中指定 logging.level 来配置它们。

```
logging.level.spring.framework=Debug
```

二十八、您将如何在微服务上执行安全测试？

您需要独立测试各个部分。有三种常见的程序：

- 代码扫描 - 确保任何代码行都没有错误并且可以复制。
- 灵活性 - 安全解决方案应该是灵活的，以便可以根据系统的要求进行调整。
- 适应性 - 安全协议应该灵活和更新，以应对黑客或安全漏洞的新威胁。

二十九、什么是幂等性？它是如何使用的？

幂等性指的是这样一种场景：您重复执行一项任务，但最终结果保持不变或类似。幂等性主要用作数据源或远程服务，当它接收一组以上指令时，它只处理一组指令。