2.1java

2.2Springboot

Spring团队决定他们希望为开发人员提供一些相对自动化配置过程的实用程序,并加快构建和部署Spring应用程序的过程,因此他们发明了Spring Boot。Spring Boot是一个实用程序项目,旨在轻松构建基于Spring,生产就绪的应用程序和服务,而不用担心。它提供了使用最小的代码/配置开箱即用的Spring Web应用程序启动和运行的最短路径。有许多特定于Spring Boot的功能,如依赖管理,自动配置和嵌入式servlet容器。

为了加速依赖关系管理过程,Spring Boot隐式地为每种类型的Spring应用程序打包所需的兼容第三方依赖项,并使用starters将它们公开给开发人员。

启动器是一组方便的依赖关系描述符,可以在应用程序中包含这些描述符。可以获得所需的所有Spring和相关技术的一站式服务,而无需搜索示例代码并复制粘贴的依赖描述符。如果想创建一个Spring Web应用程序,只需添加spring-boot-starter-web依赖项,默认情况下,这将拉出所有常用的库来开发Spring MVC应用程序,如spring-webmvc,jackson-json ,validation-api和Tomcat。换句话说,Spring Boot收集所有公共依赖项并在一个地方定义它们,并允许开发人员使用它们,而不是每次创建新应用程序时重新发明轮子。因此,pom.xml变得比传统Spring应用程序使用的小得多。

Spring Boot的另一大功能是自动配置。选择适当的启动器后,Spring Boot会尝试根据添加的jar依赖项自动配置Spring应用程序。例如,如果添加spring-boot-starter-web,Spring Boot会自动配置常见的Bean,如DispatcherServlet,ResourceHandlers,MessageSource。此外,如果使用的是spring-boot-starter-jdbc,Spring Boot会自动注册DataSource,EntityManagerFactory和TransactionManager bean,并从application.properties文件中读取连接详细信息。如果不打算使用数据库而您没有提供任何手动连接详细信息,Spring Boot将自动配置内存数据库,无论何时在其上找到H2或HSQL库,都无需进行任何进一步配置。建立路径。

默认情况下,每个Spring Boot Web应用程序都包含一个嵌入式Web服务器,请选中此选项以获取开箱即用的嵌入式servlet容器列表。开发人员无需担心设置servlet容器并在其上部署应用程序。应用程序可以使用其嵌入式服务器作为可运行的jar文件自行运行。如果需要使用单独的HTTP服务器,则只需要排除默认依赖项,Spring Boot为HTTP服务器提供单独的启动程序,以帮助使此过程尽可能简单。使用嵌入式服务器创建独立的Web应用程序不仅便于开发,而且是企业级应用程序的合法解决方案,并且在微服务领域越来越有用。能够将整个服务(例如,用户身份验证)包装在暴露API的独立且完全可部署的工件中,使得分发和部署更加快速和易于管理。

2.3mybatis

MyBatis数据映射器框架使关系数据库与面向对象的应用程序的使用变得更加容易。 MyBatis使用XML描述符将对象与存储过程或SQL语句耦合在一起。简单性是MyBatis数据映射器相对于对象关系映射工具的最大优势。要使用MyBatis数据映射器,您需要依赖自己的对象,XML和SQL。没有什么可以学到的,你还不知道。使用MyBatis Data Mapper,可以轻松获得SQL和存储过程的全部功能。

2.4shiro

Apache Shiro的首要目标是易于使用和理解。

以下是Apache Shiro可以做的一些事情:

验证用户以验证其身份

为用户执行访问控制,例如:

确定是否为用户分配了某个安全角色

确定是否允许用户执行某些操作

在任何环境中使用Session API,即使没有Web容器或EJB容器也是如此。

在身份验证,访问控制或会话生命周期内对事件做出反应。

聚合用户安全数据的1个或多个数据源,并将其全部显示为单个复合用户"视图"。

启用单点登录(SSO)功能

无需登录即可为用户关联启用"记住我"服务

...

还有更多-所有这些都集成在一个易于使用的内聚API中。

Shiro尝试为所有应用程序环境实现这些目标-从最简单的命令行应用程序到最大的企业应用程序,而不会强制依赖其他第三方框架,容器或应用程序服务器。当然,该项目旨在尽可能地融入这些环境,但它可以在任何环境中开箱即用

Apache Shiro是一个具有许多功能的综合应用程序安全框架。下图显示了Shiro关注其能量的位置,此参考手册将以类似方式组织:

Shiro针对Shiro开发团队所称的"应用程序安全的四大基石"-身份验证,授权,会话管理和加密:

身份验证:有时称为"登录",这是证明用户是他们所说的人的行为。

授权:访问控制的过程,即确定"谁"可以访问"什么"。

会话管理:即使在非Web或EJB应用程序中,也可以管理特定于用户的会话。

密码学:使用加密算法保持数据安全,同时仍然易于使用。

还有其他功能可以在不同的应用程序环境中支持和强化这些问题,尤其是:

Web支持:Shiro的Web支持API可帮助轻松保护Web应用程序。

缓存:缓存是Apache Shiro API中的第一层公民,可确保安全操作保持快速高效。

并发:Apache Shiro支持具有并发功能的多线程应用程序。

测试:存在测试支持以帮助您编写单元和集成测试,并确保您的代码按预期受到保护。

"运行方式":允许用户假定其他用户的身份(如果允许)的功能,有时在管理方案中很有用。

"记住我":记住用户在会话中的身份,因此他们只需要在必要时登录。

数据库是一个存储数据集合的独立应用程序。每个数据库都有一个或多个不同的API,用于创建,访问,管理,搜索和复制它所拥有的数据。也可以使用其他类型的数据存储,例如文件系统上的文件或内存中的大型哈希表,但是对于这些类型的系统,数据获取和写入不会那么快速和容易。

如今,我们使用关系数据库管理系统(RDBMS)来存储和管理大量数据。这称为关系数据库,因为所有数据都存储在不同的表中,并且使用主键或称为外键的其他键建立关系。

关系数据库管理系统(RDBMS)是一种软件使您能够实现包含表,列和索引的数据库、保证各种表行之间的参照完整性、自动更新索引、解释SQL查询并组合来自各种表的信息。

MySQL是一种快速,易于使用的RDBMS,可用于许多小型和大型企业。 MySQL由瑞典公司MySQL AB开发,营销和支持。由于很多很好的理由,MySQL变得如此受欢迎MySQL是在开源许可下发布的。因此无需支付任何费用。

MySQL本身就是一个非常强大的程序。它处理最昂贵和功能最强大的数据库包的大部分功能。MySQL使用着名的SQL数据语言的标准形式。MySQL可以在许多操作系统上运行,并且可以使用许多语言,包括PHP,PERL,C,C ++,JAVA等。MySQL工作非常快,即使对于大型数据集也能正常工作。MySQL对PHP非常友好,PHP是用于Web开发的最受欢迎的语言。MySQL支持大型数据库,表中包含多达5000万行或更多行。表的默认文件大小限制为4GB,但您可以将此值(如果您的操作系统可以处理它)增加到理论上的800万TB(TB)。MySQL是可定制的。开源GPL许可证允许程序员修改MySQL软件以适应他们自己的特定环境。

5.2 NOSQL及Redis数据库介绍

几乎每个应用程序都必须存储数据,并且通常需要大量快速变化的数据。直到最近,大多数应用程序使用关系数据库管理系统(简称RDBMS)存储数据,如Oracle,MySQL或PostgreSQL。然而,最近,由于需要以更有效的方式存储无模式数据NoSQL,因此出现了一种新的数据存储模式。选择是使用SQL还是NoSQL通常是设计成功应用程序的重要第一步。在选择是否使用SQL或NoSQL来存储数据时,有两件重要的事情要考虑:它的性质和使用模式。某些数据非常适合关系存储引擎,而其他数据则受益于Nois引擎(如Redis或其替代品)的无模式特性。

在Web上工作时,数据和数据模型可能会随着增加的功能和业务更新而不断变化。在关系数据库中发展模式以支持这些更改是一个痛苦的过程,特别适合非关联存储的数据示例包括事务详细信息,历史数据和服务器日志。这些通常是高度动态的,经常变化,并且它们的存储趋于快速增长,进一步加剧了调整模式以存储它们的问题。它们通常也不会感觉到"关系"。也就是说,它们中的数据不会在与其他类型数据的关系中散开。这是一个很好的迹象,表明他们可以使用RDBMS以外的东西。

衡量NoSQL适用性的另一种方法是查看是否因性能原因而发现自己的数据非规范化,并且不再受益于关系系统的某些优势,例如一致性和冗余检查。NoSQL数据库通常不提供ACID(原子性,一致性,隔离性,持久性),或者仅部分地执行它。这允许他们做出一些其他情况下无法实现的权衡。 Redis通过设计提供部分ACID合规性,因为它是单线程(保证一致性和隔离),并且如果始终配置appendfsync,则提供完全合规性,同时提供持久性。表现也可能是一个关键因素。 NoSQL数据库通常更快,特别是对于写操作,使它们非常适合大量写入的应用程序。

Redis是一个开源(BSD许可),内存数据结构存储,用作数据库,缓存和消息代理。它支持数据结构,如字符串,散列,列表,集合,带有范围查询的排序集,位图,超级日志,具有半径查询和流的地理空间索引。 Redis具有内置复制,Lua脚本,LRU驱逐,事务和不同级别的磁盘持久性,并通过Redis Sentinel提供高可用性并使用Redis Cluster自动分区。

您可以对这些类型运行原子操作,例如附加到字符串;递增哈希值;将元素推送到列表中;计算集合交集,并集和差异;或者在排序集中获得排名最高的成员。

为了实现其出色的性能,Redis使用内存数据集。根据您的使用情况,您可以通过每隔一段时间将数据集转储到磁盘或通过将每个命令附加到日志来保留它。如果您只需要功能丰富的网络内存缓存,则可以选择禁用持久性。

Redis还支持简单到设置的主从异步复制,具有非常快速的非阻塞第一次同步,自动重新连接以及在网络分割上的部分重新同步。其他功能包括:交易、发布/订阅、Lua脚本、钥匙的生存时间有限、LRU逐出钥匙、自动故障转移

Redis是用ANSI C编写的,适用于大多数POSIX系统,如Linux,\* BSD,OS X,没有外部依赖性。 Linux和OS X是Redis开发和测试的两个操作系统,我们建议使用Linux进行部署。 Redis可能在Solaris衍生系统(如SmartOS)中工作,但支持是最好的努力。 Windows版本没有官方支持,但Microsoft开发并维护了Redis的Win-64端口。

所有这些都说,尽管NoSQL感觉更灵活,但在RDBMS中存储关系数据也有很好的论据。如果您具有非常适合规范化的可预测数据,则可以获得使用关系数据存储引擎的好处。在做出决定之前,请始终查看数据。像Redis这样的NoSQL数据库速度快,容易扩展,非常适合许多现代问题。一旦加权了所有选项,在SQL(对于稳定的,可预测的,关系数据)和NoSQL(对于临时的,高度动态的数据)之间进行选择应该是一件容易的事。