



链滴

Spring Idap 2.3.1.RELEASE 目录

作者: [sucre](#)

原文链接: <https://ld246.com/article/1588995859258>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

1. 介绍
 1. 概览
 2. 传统Java LDAP VS LdapTemplate
 3. 2.2新特性
 4. 2.3新特性
 5. 2.4新特性
 6. 打包概览：你将会用到的依赖包
 7. 开工
 8. 支持：社区和项目地址
 9. 致谢
2. 简单使用
 1. 使用 AttributesMapper 进行查询、检索
 2. 构建 LDAP 查询
 3. 动态构建 Distinguished Names (译：专用名称)
 1. LdapName VS LdapNameBuilder
 2. LdapUtils
 4. 绑定和解绑
 1. 插入数据即为绑定
 2. LdapTemplate
 3. 删除即为解绑
 5. 修改
 1. 修改即重新绑定
3. 简化属性操作及DirContextAdapter
 1. 介绍
 2. 用 ContextMapper 进行检索和查询
 3. 使用 DirContextAdapter 添加和修改数据
 4. DirContextAdapter 和作为属性值的 Distinguished Names
 1. Distinguished name 会忽略空格，且不区分大小写，不能简单的从字符串相等性判断值。如一个 member 属性有个值：cn=John Doe,ou=People (注意没有空格) 调用 ctx.addAttributeValue("member", "CN=John Doe, OU=People") 会认为member有两个值，正确的使用方式：LdapUtils.newLdapName("CN=John Doe, OU=People")
 2. LdapNameBuilder
4. Object-Directory Mapping (ODM)
 1. 介绍：对照ORM框架，spring ldap 提供了 LdapOperators
 2. 注解：@Entry, @Id, @Attribute, @DnAttribute, @Transient
 3. ODM 和 专用名词的属性
5. 增强LDAP查询
 1. LDAP Query Builder 参数：介绍查询时有用的参数，让查询更灵活
 2. 过滤条件：就是查询条件
 3. 硬编码过滤：适用于spring不支持的查询条件
6. 配置
 1. 介绍：推荐使用xml配置
 2. ContextSource 配置及其参数
 1. DirContext 认证：使用DirContext通常需要认证
 1. spring提供另种TLS认证方式：DefaultTlsDirContextAuthenticationStrategy：支持优关闭TLS通道，这在使用native-poolnative-pooling时很重要ExternalTlsDirContextAuthenticationStrategy
 2. 使用 AuthenticationSource 实现自定义 主题和凭证管理 (Custom Principal and Credentials Management)
 2. LDAP池
 3. 增强 ContextSource 配置：自定义环境变量
 3. LdapTemplate 配置
 4. 获取基础LDAP路径的引用 (base LDAP path)：以后所有的操作都相对这个path
7. Spring LDAP 仓库

1. 概述已内部支持 [spring data ldap](<https://docs.spring.io/spring-data/data-commons/doc/1.6.1.RELEASE/reference/html/repositories.html>)由于LDAP协议规定，不支持分页和排序
2. QueryDSL 的支持
8. 连接池
 1. 介绍LDAP的连接池支持参考： [Java LDAP pooling support](<http://java.sun.com/products/ndi/tutorial/ldap/connect/pool.html>)
 2. DirContext 验证
 3. 连接池配置
 4. 连接池Pool2配置可以学习下common-pool 和 common-pool2 的区别
 5. 配置
 1. 校验的配置
 6. 已知的问题
 1. 自定义认证
9. 添加缺少的重载API方法
 1. 实现自定义搜索方法
 2. 实现其他的自定义Context方法
10. 处理DirContext
 1. 自定义 DirContext 前置/后置处理
 2. 实现请求处理器DirContextProcessor
 3. 查询结果分页目前一些LDAP服务已经支持了 PagedResultsControl。在使用分页的时候，必使用cookie保持请求轨迹，以此实现分页请求调用，这里就用到了上一节讲到的DirContext前置/后处理。PagedResultsDirContextProcessor
11. 事务
 1. 介绍spring LDAP提供了客户端的补偿事务
 2. 配置
 3. JDBC事务集成
 4. LDAP补偿事务解析
 1. 重命名策略
12. 用spring ldap 实现用户认证
 1. 基本认证
 2. 在基本认证上做一些操作
 3. 过时的认证方法
 4. 使用 spring security
13. LDIF解析
 1. 介绍LDAP Directory Interchange Format (LDIF)IdifParser
 2. 对象表现形式: LdapAttribute and LdapAttributes
 3. 解析器 ParserSeparatorPolicy: 建立哪些行需要组装成属性的机制AttributeValidationPolic: 确保在解析之前属性能够正确的构造Specification: 提供一个机制: 组装后, 哪些对象构造能被验
 4. Schema校验
 5. 集成 spring batchorg.springframework.ldap.ldif.batch
14. 工具
 1. 多值属性的增量检索如果一个属性有超过1500个值, Active Directory一般会拒绝一次返回
15. 测试
 1. 使用内嵌服务spring-ldap-test 提供基于 ApacheDS 或 UnboundID 的内嵌服务器[ApacheDS (<https://directory.apache.org/apacheds/>)[UnboundID](<https://www.ldap.com/unboundid-lda-sdk-for-java>)LdapTestUtils
 2. ApacheDS (directory server)
 3. UnboundID

Version 2.3.1.RELEASE

Last updated 2017-01-19 15:35:09 +00:00

