Phoenix操作Hbase

> 整个架构基于mybatis-plus
>
> PhoenixBaseMapper复写了insert以支持upsert
>
> AbstractPhoenixServiceImpl实现了upsertBatch
>
> DefaultPhoenixConfiguration 预留了hadoopLogin()方法（该方法现在已经实现数据组相关认证），如有需要可以通过复写该方法，实现自定义hadoop用户认证
>
> krb5.conf需放置到项目中src/main/resouces/下
>
> hadoop.keytab需放置到项目中src/main/resouces/下

添加依赖

```
        <!--  -->
        <dependencyManagement>
                <dependencies>

                        <dependency>
                                <groupId>com.acca</groupId>
                                <artifactId>araf-phoenix-spring-boot-starter</artifactId>
                                <version>${araf.version}</version>
                        </dependency>

                </dependencies>
        </dependencyManagement>




        <!-- phoenix -->
        <dependency>
                        <groupId>com.acca</groupId>
                        <artifactId>araf-phoenix-spring-boot-starter</artifactId>
        </dependency>
```

使用示例

添加PhoenixConfiguration

```
/*
 * Copyright (c) ACCA Corp.
 * All Rights Reserved.
 */
package com.acca.opra.image.configuration;

import araf.phoenix.config.DefaultPhoenixConfiguration;
import lombok.extern.slf4j.Slf4j;
import org.mybatis.spring.annotation.MapperScan;
import org.springframework.context.annotation.Configuration;

/**
 * PhoenixConfiguration.
 *
 * @author Li MingYi, 20201201
 * @version Araf v2.0
 */
@Configuration
@MapperScan({ "com.acca.opra.image.mapper" })
@Slf4j
public class PhoenixConfiguration extends DefaultPhoenixConfiguration {

}
```

修改application.yml

```
spring:
  datasource:
    driver-class-name: org.apache.phoenix.jdbc.PhoenixDriver
    url: jdbc:phoenix:10.1.18.208,10.1.18.209,10.1.18.210:2181
    username:
    password:
    type: com.alibaba.druid.pool.DruidDataSource
    druid:
      initial-size: 5
      min-idle: 5
      max-active: 10
      connect-properties: { hbase.master.kerberos.principal: hbase
/_HOST@ACCA,hbase.regionserver.kerberos.principal: hbase/_HOST@ACCA,hbase.
security.authentication: kerberos, hadoop.security.authentication:
kerberos,phoenix.schema.isNamespaceMappingEnabled: true }




#krb5.conf  & hadoop.keytab
hbase:
  keyTabPath: E:\\source_acca\\ORRA-GIT\\opra-nbi-image-api\\opra-nbi-
image-api-server\\src\main\\resources\\
```

创建domainmodel

```
/*
 * Copyright (c) ACCA Corp.
 * All Rights Reserved.
 */
package com.acca.opra.image.domainmodel;

import com.baomidou.mybatisplus.annotation.TableField;
import com.baomidou.mybatisplus.annotation.TableId;
import com.baomidou.mybatisplus.annotation.TableName;

/**
 * ImgTicket
 *
 * @author Li MingYi, 20201204
 * @version OPRA v1.0
 */
@TableName("OPRACA.IMG_TICKET")
public class ImgTicket {

    @TableId("IMG_KEY")
    private String imgkey;
    @TableField("INFO")
    private String info;
    @TableField("DATA_SOURCE")
    private String dataSource;

}
```

创建mapper

```java
/*
 * Copyright (c) ACCA Corp.
 * All Rights Reserved.
 */
package com.acca.opra.image.mapper;

import araf.phoenix.mapper.PhoenixBaseMapper;
import com.acca.opra.image.domainmodel.ImgTicket;
import org.apache.ibatis.annotations.Param;
import org.apache.ibatis.annotations.Select;

/**
 * ImgTicketMapper
 *
 * @author Li MingYi, 20201207
 * @version OPRA v1.0
 */
public interface ImgTicketMapper extends PhoenixBaseMapper<ImgTicket> {

    @Select("select INFO,DATA_SOURCE,IMG_KEY from OPRACA.IMG_TICKET where IMG_KEY in (select IMG_KEY from OPRACA.IMG_TICKET_INDEX where TICKET_NO=#{ticketNo})")
    ImgTicket getByTicketNo(@Param("ticketNo") String ticketNo);
}
```

创建Service

```
/*
 * Copyright (c) ACCA Corp.
 * All Rights Reserved.
 */
package com.acca.opra.image.service;

import araf.phoenix.service.impl.AbstractPhoenixServiceImpl;
import com.acca.opra.image.domainmodel.ImgTicket;
import com.acca.opra.image.mapper.ImgTicketMapper;
import org.springframework.stereotype.Service;

/**
 * PhoenixService
 *
 * @author Li MingYi, 20201204
 * @version OPRA v1.0
 */
@Service
public class ImgTicketService extends
AbstractPhoenixServiceImpl<ImgTicketMapper, ImgTicket> {

}
```

测试类

```
package com.acca.opra.image.service;

import com.acca.opra.image.boot.AppServer;
import com.acca.opra.image.domainmodel.ImgTicket;
import com.acca.opra.image.mapper.ImgTicketMapper;
import org.junit.Test;
import org.junit.runner.RunWith;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.boot.test.context.SpringBootTest;
import org.springframework.test.context.junit4.SpringRunner;

/**
 * PhoenixServiceTest
 *
 * @author Li MingYi, 20201204
 * @version OPRA v1.0
 */
@RunWith(SpringRunner.class)
@SpringBootTest(classes = { AppServer.class })
public class ImgTicketServiceTest {

    @Autowired
    ImgTicketMapper imgTicketMapper;

    @Autowired
    ImgTicketService imgTicketService;

    @Test
    public void getByTicketNo() {
        ImgTicket imgTicket = imgTicketMapper.getByTicketNo("1451566870");
        System.out.println(imgTicket);
    }

    @Test
    public void getById() {
        ImgTicket imgTicket = imgTicketService.getById
("078665154178420190801D");
        System.out.println(imgTicket);
    }

}
```