# Data-Translate翻译工具使用手册

1. 工具简介
   1. 简介

springboot,springmvc数据翻译工具，即支持通过注解开启数据翻译数据属性功能，并把翻译结果存到数据库。

* 1. 源码地址

<https://gitee.com/asgards/data-translate>

<https://github.com/asgard2023/data-translate>

* 1. maven

<dependency>

    <groupId>cn.org.opendfl</groupId>

    <artifactId>data-translate</artifactId>

    <version>1.5</version>

</dependency>

1. 功能说明
   1. 功能特性

支持springboot,springmvc

支持mysql,postgresql等

支持easyui,jqgrid,layui前端对功能翻译的修改

支持java注解使用

支持百度,有道，google等翻译api

支持本地缓存翻译结果guava 1分钟

支持redis缓存翻译结果20分钟(数据修改会即时清除redis缓存)

支持redis取上次缓存翻译结果并同步到本次缓存，即热点数据越用越快

支持用翻译api进行初始翻译，并保存到数据库。

支持对翻译结果进行修改，并以修改后的结果为准。

建议使用数字做id

* 1. 使用限制

PO或VO类名不要重复，按类名默认entityPO做翻译的key。

首次翻译时是调用翻译api进行翻译，行数多会比较慢。

查询数据每页行数不宜过多，30行以下为宜。

支持的语言见：LangType.java类

翻译时可并发多线程调用翻译api，可能会受翻译供应商QPS限制。

1. 功能使用
   1. 引入依赖

<dependency>

    <groupId>cn.org.opendfl</groupId>

    <artifactId>data-translate</artifactId>

    <version>1.5</version>

</dependency>

* 1. Java使用
     1. 在要翻译对象PO或VO类加翻译注解类

//一个类可以有一个或多个属性支持内容翻译

/\*\*

 \* 支持翻译的PO类，加上@TranslateType

 \*/

@Data

@TranslateType

**public** **class** DflUserPo **implements** Serializable {

**private** Integer id;

    /\*\*

     \* 昵称，要翻译的属性

     \*/

    @TranslateField

**private** String nickname;

    /\*\*

     \* 描述信息，要翻译的属性

     \*/

    @TranslateField

**private** String descs;

}

* + 1. 在返回的list调用翻译功能

示例：<http://translate-demo.opendfl.org.cn/dflUser/listByLang?pageSize=20&pageNum=1&lang=en>

@RestController

@RequestMapping("dflUser")

**public** **class** DflUserController{

    @RequestMapping(value = "/listByLang", method = {RequestMethod.POST, RequestMethod.GET})

**public** List<DflUserPo> listByLang(HttpServletRequest request, DflUserPo entity, MyPageInfo<DflUserPo> pageInfo) {

        String lang = request.getHeader(RequestParams.LANG);//lang=en,jp等，主语言zh不翻译

**if** (StringUtils.isBlank(lang)) {

            lang = request.getParameter(RequestParams.LANG);

        }

        logger.debug("-------listByLang--lang={}", lang);

**this**.pageSortBy(pageInfo);

        pageInfo = dflUserBiz.findPageBy(entity, pageInfo, **this**.createAllParams(request));

**if** (StringUtils.isNotBlank(lang)) {

            //方式一，调用接口翻译处理

            TranslateUtil.transform(lang, pageInfo.getList(), **true**);

        }

**return** pageInfo.getList();

    }

}

* + 1. 注解@Translate

示例：<http://translate-demo.opendfl.org.cn/dflUser/listByLang2?pageSize=20&pageNum=1&lang=en>

**import** cn.org.opendfl.translate.dflsystem.translate.annotation.Translate;

@RestController

@RequestMapping("dflUser")

**public** **class** DflUserController {

    //方式二，通过注解翻译处理

    @Translate

    @RequestMapping(value = "/listByLang2", method = {RequestMethod.POST, RequestMethod.GET})

**public** List<DflUserPo> listByLang2(HttpServletRequest request, DflUserPo entity, MyPageInfo<DflUserPo> pageInfo) {

**this**.pageSortBy(pageInfo);

        pageInfo = dflUserBiz.findPageBy(entity, pageInfo, **this**.createAllParams(request));

**return** pageInfo.getList();

    }

}

* + 1. 翻译特性-属性替换

Java翻译会开启翻译替换，即把属性的原文替换成译文，

比如：name: ‘测试’，变成：name: ‘test’

* 1. Jqgrid使用

示例：<http://translate-demo.opendfl.org.cn/pages/dflUser.html>

* + 1. List返回dicts.typeInfo

**public** MyPageInfo<DflUserPo> queryPage(HttpServletRequest request, DflUserPo entity, MyPageInfo<DflUserPo> pageInfo) {

    pageInfo = dflUserBiz.findPageBy(entity, pageInfo, **this**.createAllParams(request));

    //用于返回typeInfo(主要是翻译的属性名transFields)

    pageInfo.setDicts(TranslateUtil.dictMap(DflUserPo.**class**));

    //启动内容未翻译自动翻译

    TranslateUtil.transformLangsByTrnasType(request, pageInfo.getList());

**return** pageInfo;

}

* + 1. 引入js

<script src="../../js/common.translate.js"></script>

* + 1. 增加翻译属性列并支持可编辑

//下面通过trans\_jqGridCells自动把翻译的信息补到翻译列上。

{name: 'descs', index: 'descs',editable:**true**, width: 100, editrules:{required:**true**}},

{name: 'descsEn', index: 'descsEn',editable:**true**, width: 100, editrules:{required:**true**}},

{name: 'descsTw', index: 'descsTw',editable:**true**, width: 100, editrules:{required:**true**}},

{name: 'descsJa', index: 'descsJa',editable:**true**, width: 100, editrules:{required:**true**}}

* + 1. 加载系统配置的翻译语言类型

//获取翻译语言类型en,tw,ja等

trans\_getTypeDists();

* + 1. 列表数据加载成功后渲染之前

loadComplete: function (r){

    var list=r.list;

    var dicts=r.dicts;

    var typeInfo;

**if**(dicts){

        typeInfo=dicts.typeInfo;

    }

    trans\_jqGridCells('table\_list\_1', list, transTypeCode, typeInfo);

}

* + 1. 保存数据时同时保存翻译内容的修改

//该数据全部翻译属性每种语言全部保存一遍，翻译内容未修改时，最后不会执行数据更新

trans\_saveLocalLanguage(result.data, postdata, transFields);

* 1. Easyui使用

示例：<http://translate-demo.opendfl.org.cn/pages/dflRole.html>

功能上同

列表数据加载成功后渲染之前改成以下方式

**var** typeInfo;

**if**(data.dicts && data.dicts.typeInfo){

    typeInfo=data.dicts.typeInfo;

}

trans\_loadSuccessIds(data.rows, transTypeCode, typeInfo);

* 1. Layui使用

示例：<http://translate-demo.opendfl.org.cn/pages/dflResource.html>

功能同上

列表数据加载成功后渲染之前改成以下方式

**function** parseDataFn(res){

    console.log('parseDataFn');

**var** typeInfo;

**if**(res.dicts && res.dicts.typeInfo){

        typeInfo=res.dicts.typeInfo;

    }

    trans\_loadSuccessIds(res.data, transTypeCode, typeInfo);

}