domain-tool REST API接口说明

<http://www.boluogan.com>

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 修改人 | 发布日期 | 修改内容 |
| 0.1 | 老林 | 2015/1/8/ | 初始版本 |
| 0.1 | 老林 | 2015/1/12 | 更新rest api文档 |
|  |  |  |  |
|  |  |  |  |

目录

简介 2

默认根路径 2

REST API说明 3

全部根后缀 3

可以解析结果的根后缀 3

单个域名查询 4

域名批量查询接口 5

websocket接口调用说明 5

代表websocket客户端的JS全局变量 5

接口的连接和订阅 5

发送批量查询请求 6

断开接口连接 6

websocket客户端程序设计（建议） 6

批量查询的返回数据说明 7

保存查询结果 7

查看历史查询结果 8

# 1.简介

接口程序基于spring-boot开发，用Maven构建，提供restful风格的api接口。

项目代码在<https://github.com/boluogan/domain/tree/master/domain-tool>

# 2.默认根路径

<http://localhost:18080/>

# 3.REST API说明

## 3.1全部根后缀

本接口列出全部可以查询的域名根后缀

|  |  |
| --- | --- |
| 元素 | 说明 |
| 接口地址 | tlds/root/all |
| 调用类型 | GET |
| 参数 | 无参数 |
| 返回 | Json数据，格式如下:  ["se","as","edu","va","dm","re","in","fo","br","pl","ie",  "vg","jobs","gl","th","com","cn","coop","ua","lk","sk",  "cd","org","ru","sh","cx","at","bg","pt","me","na","cc",  "it","si","tf","ee","jp","my","tm","ch","bz","il","is",  "ly","am","nl","tc","tl","tk","eu","info","li","ag","be",  "au","pro","kr","no","ws","name","hu","int","cat","tv",  "mx","la","sg","museum","lt","fr","ca","vc","ae","us",  "ck","mc","af","nu","aero","sa","hm","iq","mobi","nf",  "dk","lv","mil","cl","ms","sc","ac","gd","tw","biz",  "uk","tr","lu","ro","st","mu","net","de","sm","fi",  "nz","hk","tp","to","ki","gs","gov","cz","asia"] |

## 3.2可以解析结果的根后缀

|  |  |
| --- | --- |
| 元素 | 说明 |
| 接口地址 | tlds/root/parseable |
| 调用类型 | GET |
| 参数 | 无参数 |
| 返回 | Json数据，格式如下:  ["uk","com","hk","us","net","in","biz","info","cn","tw","tv","name","me","edu","cc","cd"] |

这些后缀的域名，其 whois查询结果字符串可以转成DomainWhoisInfo和WhoisRecord对象。

## 3.3单个域名查询

查询单个域名的whois信息。

|  |  |
| --- | --- |
| 元素 | 说明 |
| 接口地址 | whois/${domainname} |
| 调用类型 | GET |
| 参数 | 域名（不带www，如:boluogan.com） |
| 返回 | Json数据，格式如下:  {"status":"SUCCESS","message":{"sid":1,"domainname":"abc.hk","registerStatus":"已注册","createDate":"2010-09-03 00:00:00","expirationDate":"2015-09-03 00:00:00","registrant":null,"registrantEmail":"domain@mail.im","registrar":"Hong Kong Domain Name Registration Company Limited"}} |

## 3.4域名批量查询接口

这是一个websocket接口，需要连接websocket，订阅消息，然后在需要查询的时候发送消息到websocket接口，等待接口把查询信息推送回来。

具体调用可以参考示例页面：

<http://localhost:18080/scan.html>

### 3.4.1websocket接口调用说明

在示例程序中，用到了sockjs-0.3.4.js和storm.js开源框架。

### 3.4.2代表websocket客户端的JS全局变量

var stompClient = null;

### 3.4.3接口的连接和订阅

function connect() {

var socket = new SockJS('/domainscan');

stompClient = Stomp.over(socket);

stompClient.connect({}, function(frame) {

setConnected(true);

console.log('Connected: ' + frame);

stompClient.subscribe('/topic/bulkquery', function(responseData){

//var responseDataBody = JSON.parse(responseData.body);

//showMessage(responseData.body);

});

});

}

### 3.4.4发送批量查询请求

stompClient.send("/app/domainscan", {}, JSON.stringify([{"sid":1,"domainname":"boluogan.com"},{"sid":2,"domainname":"pingguogan.com"}]));

### 3.4.5断开接口连接

function disconnect() {

stompClient.disconnect();

//setConnected(false);

console.log("Disconnected");

}

### 3.4.6websocket客户端程序设计（建议）

建议查询页面在加载完毕之后，就连接websocket。

批量查询事件的出发按钮可以在非连接状态之下设置为不可连接。

### 批量查询的返回数据说明

发送批量请求之后，推送返回json数据：

{"status":"BEGIN","message":"开始查询...."}

每查询完一个域名，讲查询到的whois信息推送回来：

{"status":"QUERYING","message":{"sid":1,"domainname":"boluogan.com","registerStatus":"已注册","createDate":"2014-06-11 00:00:00","expirationDate":"2015-06-11 00:00:00","registrant":null,"registrantEmail":null,"registrar":"ENAME TECHNOLOGY CO., LTD."}}

查询结束后，返回(status为Done，表示查询结束。)：

{"status":"DONE","message":"批量查询结束."}

## 3.5保存查询结果

为此次域名查询指定一个名称，并将查询结果保存。

|  |  |
| --- | --- |
| 元素 | 说明 |
| 接口地址 | savequeries |
| 调用类型 | POST |
| 参数 | Json数据，数据格式参考：  {“name”:”以daikuan为开头的三拼COM查询”,  “data”:[{"sid":1,"domainname":"abc.hk","registerStatus":"已注册","createDate":"2010-09-03 00:00:00","expirationDate":"2015-09-03 00:00:00","registrant":null,"registrantEmail":"domain@mail.im","registrar":"Hong Kong Domain Name Registration Company Limited"}  ]} |
| 返回 | Json数据，格式如下:  {"status":"SUCCESS","message":"处理成功"} |

## 3.6查看历史查询记录

列出已保存的历史查询记录（3.5一节所说接口所保存的全部数据）。

|  |  |
| --- | --- |
| 元素 | 说明 |
| 接口地址 | queries |
| 调用类型 | GET |
| 参数 | 无参数 |
| 返回 | Json数据，格式如下:  [{"name":"以daikuan为开头的三拼COM查询","url":"/datas/20141217203508.json"  }] |