

金笛短信中间件 web 版

开发接口说明

数据库接口说明

配置完数据源之后,中间件会自动创建 smsserver_in、smsserver_out、smsserver_calls 三张表, smsserver_out 表用于存放待发送的短信以及短信的发送结果,将需要发送的短信插入到该表,中间件会自动发送,并将发送结果记录到该表。

Smsserver_in 表,中间件收到短信以及状态报告之后,会将其存入该表。

Smsserver_calls 表,中间件接到电话之后会将其存入该表。

数据库接口支持的开发语言: **java**、**C**、**C++**、**C#**、**PHP**、**Ruby**、**VB** 等一切可以操作数据库的编程语言。

数据库表结构说明:

Smsserver_in 表:

字段名	类型	属性	描述
id	INT	自动	primary key.
process	INT	NOT NULL	预留备用
originator	CHAR(16)	NOT NULL	发件人
type	CHAR(1)	NOT NULL	短信类型: "I"普通短信, "S"短信状态报告
encoding	CHAR(1)	NOT NULL	编码格式: "7" for 7bit, "8" for 8bit and "U" for Unicode/UCS2
message_date	DATETIME	NOT NULL	短信发送日期时间
receive_date	DATETIME	NOT NULL	接收日期时间
text	CHAR(xxx)	NOT NULL	正文
original_ref_no	CHAR(64)	NULL	用于短信状态报告: 此号码对应于外发的短信序号
original_receive_date	DATETIME	NULL	用于短信状态报告: 对方收到短信的日期时间
gateway_id	CHAR(64)	NOT NULL	通道标识,用于标识是哪个通道收到的短信

Smsserver_out 表结构说明

字段名	类型	属性	描述
id	INT	自动	primary key.
type	CHAR(1)	NOT NULL	短信类型. "O"普通短信, "W" wappush
recipient	CHAR(16)	NOT NULL	收件人手机号码, 前面不要加 "+" 或者 0
text	CHAR(xxx)	NOT NULL	短信正文
wap_url	CHAR(xxx)	NOT NULL for WAP SI	Wap 短信 URL 地址
wap_expiry_date	DATETIME	NULL	Wap 短信有效期, 默认7天。
wap_signal	CHAR(1)	NULL	"N" NONE, "L" LOW, "M" MEDIUM, "H" HIGH, "D" DELETE
create_date	DATETIME	NOT NULL, Default value: current date/time	记录插入数据库的时间
originator	CHAR(16)	NOT NULL, Default value: "	发件人正常情况下为空
encoding	CHAR(1)	NOT NULL, Default value: '7'	编码格式: "7" 7bit, "8" 8bit and "U" Unicode/UCS2。发送中文短信请设为 U。
status_report	INT	NOT NULL, Default value: 0	状态报告: 设为 1需要短信状态报告, 默认为0。
flash_sms	INT	NOT NULL, Default value: 0	闪信, 设为1 发送闪信, 默认为0。
src_port	INT	NOT NULL, Default value: -1	源端口 (对彩铃有效)
dst_port	INT	NOT NULL, Default value: -1	目的端口 (对彩铃有效)
sent_date	DATETIME	NULL	发送时间, Jdsmsserver 发送时会更新这个时间。
ref_no	CHAR(64)	NULL	短信对应 ID
priority	INT	NOT NULL, Default value: 0	优先级: 低: 负数 普通: 0 高: 正数
status	CHAR(1)	NOT NULL, Default value "U"	短信状态: "U":未发送, "Q": 排队中, "S": 已发送, "F": 失败
errors	INT	NOT NULL,	重发次数

gateway_id	CHAR(64)	Default value: 0 NOT NULL, Default value is the star character	短信通道标识, 可指定发送通道。为空 由短信服务器自动选择。
user_id	CHAR(20)		用户标识, 主要是通过界面发送时由哪 个用户提交的

Smsserver_calls 表结构说明:

字段名	类型	属性	描述
id	INT	自动	primary key.
call_date	DATETIME	NOT NULL	电话呼入时间
gateway_id	CHAR(64)	NOT NULL	通道标识
caller_id	CHAR(16)	NOT NULL	对方号码

功能测试

进入到控制台管理界面, 选择短信发送界面, 给您自己的手机发送一条短信, 如果收到表示测试成功。也可以直接往数据库里边插入一条记录, 使用如下 SQL 语句
 insert into smsserver_out (type,recipient,text,encoding,
 create_date) values ('O','13601019694','jdsmsserver 数据库发送成功!',
 U',now());

URL 收发接口说明

金笛短信中间件 web 版提供了三个 url 分别用于收发短信和查询短信发送结果。

短信发送 URL:

[http://ip:8060/send?password=\[密 码 \]&text=\[内 容 \]&recipient=\[手 机 号\]&encoding=U&gateway=*](http://ip:8060/send?password=[密 码]&text=[内 容]&recipient=[手 机 号]&encoding=U&gateway=*)

发送参数说明

ip: 安装 web 中间件所在服务器的 ip。

password: 发送密码, 默认是 1。

text: 发送内容, 该字段实际设置的是您要发送的内容的 utf8 编码。

recipient: 对方手机号码, 群发时用逗号分割。

encoding: 字符编码集, 设置为 U 发送中文短信。

gateway: 从哪一个通道发出, * 代表不指定, 由系统分配。

可选发送参数说明:

priority: 优先级, 低: 负数 普通: 0 高: 正数

from: 发送者名称, 部分运营商支持, 只针对 GSM modem

返回的结果说明:

举例如下:

成功的例子:

<?xml version="1.0" encoding="UTF-8"?>

```

<send>
  <error>0</error>
  <message_status>UNSENT</message_status>
  <ref_no>221</ref_no>
  <msg_id>2</msg_id>
  <gateway>modem1</gateway>
</send>

```

返回错误码说明

属性	属性说明	描述
error	错误代码	0 表示发送成功。-9000: 参数非法, -9001: 缺少参数, -9002: 发送错误, -9003: 密码错误, -9999: 发送终止
message_status	短信的发送状态	UNSENT (未发送)、SENT 已发送、FAILED 发送失败
ref_no	短信对应 ID	短信对应 ID
msg_id	NOT NULL	短信提交到库后返回的对应 msgid
gateway	实际发送短信的通道	实际发送短信的通道

失败的例子:

```

<?xml version="1.0" encoding="UTF-8"?>
<send>
  <error>-9002</error>
  <error_description>Message not sent.</error_description>
  <message_status>FAILED</message_status>
  <failure_cause>UNKNOWN</failure_cause>
  <ref_no>null</ref_no>
  <gateway>*</gateway>
</send>

```

error_description: 错误描述

failure_cause: 导致发送失败的原因, 有以下几种情况:

BAD_NUMBER 号码格式有误

BAD_FORMAT 消息格式不正确

GATEWAY_FAILURE Generic gateway failure - 临时错误, 稍后重试.

NO_ROUTE 设备没启动或没有设备 .

UNKNOWN 未知错误.

短信发送状态查询 URL:

[http://ip:8060/sendstatus?msgid=\[msgid\]](http://ip:8060/sendstatus?msgid=[msgid])

发送参数说明

msgid: 短信提交成功后返回的 msgid。

返回的结果说明:

属性	属性说明	描述
status	发送状态	S 代表已发送, F 发送失败, U 未发送, Q 正在发送

```
<?xml version='1.0' encoding="UTF-8"?>
```

```
<status>S</status>
```

status:S 代表已发送,F 发送失败,U 未发送,Q 正在发送

短信接收 URL:

[http://ip:8060/read?password=\[密码\]&gateway=modem\[设备序号\]](http://ip:8060/read?password=[密码]&gateway=modem[设备序号])

接收参数说明

password: 接收密码,默认为 1。

Gateway: 短信模块,用于从指定模块接收短信,如果没有指定则从所有模块读取短信。

返回的结果说明:

```
<?xml version='1.0' encoding="UTF-8"?>
```

```
<messages>
```

```
  <error>0</error>
```

```
  <message>
```

```
    <message_type>Inbound</message_type>
```

```
    <message_date>2012-03-22T14:45:11 08:00</message_date>
```

```
    <originator>8615101091940</originator>
```

```
    <text>
```

```
    <![CDATA[
```

```
    金笛短信中间件web版httpserver接口
```

```
    ]]>
```

```
  </text>
```

```
  <encoding>ENCUCS2</encoding>
```

```
  <gateway>modem1</gateway>
```

```
</message>
```

```
</messages>
```

属性	属性说明	描述
message	对应于一条接收到的短信。它包含了许多子元素	对应于一条接收到的短信。它包含了许多子元素 INBOUND: 接收到的短信 OUTBOUND: 发送出去的短信
message_type	短信类型	STATUSREPORT: 状态报告 WAPSI: WAP_PUSH 短信 UNKNOWN: 未知类型
message_date	发送方发送该短信的时间	发送方发送该短信的时间
originator	发送方手机号码	发送方手机号码
text	短信内容	<![CDATA[]]> 这一段xml代码, 请自行截取掉, 短信内容采用的UTF-8, 您在接收到短信内容之后, 需要将其转换为GB2312编码(中文编码格式), 例子中是已经解码后的短信内容。
encoding	短信内容的编码方式	短信内容的编码方式
gateway	短信模块	收到该短信的模块

在 develop 文件夹有 java 的关于 httpserver 接口收发短信的历程, 可以参考一下。

回调功能说明

对收到的短信进行实时处理,比如通过设备发一条短信,对方收到后回复一条,我需要通过短信内容和发送人,并能进行相应的处理:

1:在配置文件 JDSMSServer.conf 中把这几行注释去掉,

```
#interface.2=http1,HttpPush
#http1.get_url=http://127.0.0.1:8090/test.jsp?gatewayId=%gatewayI
#d%&text=%text%&originator=%originator%
#http1.method=GET
```

其中 http1.get_url=http://127.0.0.1:8090/test.jsp 换成你的 web 服务器的地址和相应的页面,最后重新启动一下服务

其中 Console 文件夹中有对应的 test.jsp 文件可以参考一下。

邮件接口说明

往一个指定邮箱发送短信,内容的第一行是手机号码,第二行开始为内容(如果要群发的话,第一行手机号码用;分割,由中间件负责把这封邮件封装成短信发送给邮件内容第一行的这个手机号码

使用邮件接口的前提条件:

在邮箱里开启 pop3 和 smtp 服务,允许客户端读取短信
邮件内容不能带附件

1:在配置文件 JDSMSServer.conf 中把这几行注释去掉,

```
# Here is an example of a Email interface.
#interface.2=email1,Email
#email1.mailbox_protocol=pop3
#email1.mailbox_host=pop.qq.com
#email1.mailbox_port=110
#email1.mailbox_user=1843933555@qq.com
#email1.mailbox_password=11
#email1.mailbox_name=INBOX
#email1.to=1843933555@qq.com
#email1.from=1843933555@qq.com
#email1.smtp_host=smtp.qq.com
#email1.smtp_port=25
#email1.smtp_user=1843933555@qq.com
#email1.smtp_password=11
#email1.message_subject=SMS from %originator%
#email1.message_body=emailbody.html.tpl
```

配置好相关参数,重新启动一下服务