

Wind R 数据及交易接口

Version 1.1

修订时间：2014.02.12

上海万得信息技术股份有限公司

Shanghai Wind Information Co., Ltd.

地 址 上海浦东新区福山路 33 号建工大厦 9 楼

邮编 Zip 200120

电话 Tel (8621) 6888 2280

传真 Fax (8621) 6888 2281

主页 <http://www.wind.com.cn>

版本历史

时间	更新信息	备注
2013.07.09	初版	
2013.08.15	增加交易接口和条件选股	
2014.02.12	增加组合上传接口	

目录

1	WINDR 接口说明.....	1
1.1	WINDR 接口概述	1
1.2	WINDR 接口安装	1
1.2.1	WindR 对系统环境要求	1
1.2.2	R 环境安装.....	2
1.2.3	正常 WindR 接口安装	1
1.2.4	特殊安装 WindR 方式	3
1.3	WINDR 接口向导界面	3
1.4	WINDR 获取帮助途径	5
1.4.1	本用户手册	5
1.4.2	R 里面的帮助文档	5
1.4.3	量化交易群和 R 语言交流群	7
1.5	WINDR 接口相关规范	1
1.5.1	命令区分大小写，且“w.”不能省略	1
1.5.2	单字节码和双字节码的问题	1
1.5.3	品种、指标、参数等引号内的部分不区分大小写	1
1.5.4	参数支持数组输入	1
1.5.5	时间、日期支持 R 语言的时间、日期格式	2
1.5.6	参数中有缺省值的可以不用输入	2
1.5.7	可以带参数名输入	2

1.5.8 Showblank 参数.....	2
1.5.9 交易接口中 Showfields 参数.....	3
1.5.10 ErrorCode 定义	3
2 WIND R 插件命令说明	6
2.1 LIBRARY (WINDR): 装载 WINDR 包	6
2.2 ?WINDR: 启动 WINDR 帮助文档	6
2.3 W.START: 启动 WINDR.....	6
2.4 W.STOP: 停止 WINDR.....	7
2.5 W.MENU: 显示导航界面	7
2.6 W.ISCONNECTED: 判断是否已经登录	8
2.7 W.CANCELREQUEST: 取消订阅	8
2.8 W.ASDATETIME: 把数字化时间格式转换成 R 语言时间格式	9
2.9 W.WSD: 获取历史序列数据.....	9
2.10 W.WSI: 获取分钟数据	10
2.11 W.WST: 获取日内 TICK 级别数据	11
2.12 W.WSS: 获历史截面数据	12
2.13 W.WSQ: 获取和订阅实时行情数据	13
2.14 W.WSET: 获取板块、指数等成分数据	14
2.15 W.WEQS: 获取条件选股结果.....	15
2.16 W.WPF: 获取资产管理、组合管理数据	15
2.17 交易相关函数.....	1
2.17.1 w.tlogon 交易登录	1

2.17.2	w.tlogout 交易登出	1
2.17.3	w.torder 委托下单	2
2.17.4	w.tcancel 撤销委托	3
2.17.5	w.tquery 交易查询	4
2.18	W.TDAYS, W.TDAYSOFFSET, W.TDAYSCOUNT: 日期函数	5
2.18.1	w.tdays: 返回区间内的日期序列	5
2.18.2	w.tdaysoffset: 返回某个偏移值对应的日期	6
2.18.3	w.tdayscount: 返回某个区间内日期数量	6
3	WINR 插件函数体说明	8
3.1	日期序列 (WSD)	8
3.2	历史截面数据 (WSS)	10
3.3	分钟序列 (WSI)	10
3.4	日内跳价 (WST)	11
3.5	实时数据 (WSQ)	12
3.6	数据集 (WSET)	13
3.7	条件选股 (WEQS)	13
3.8	资管函数 (WPF)	13
3.9	组合上传函数 (WUPF)	15
3.10	交易函数	17
3.10.1	登录 (tlogon)	17
3.10.2	登出 (tlogout)	18
3.10.3	下单 (torder)	18

3.10.4	撤单 (tcancel)	20
3.10.5	查询 (tquery)	20
3.11	日期函数	22
3.11.1	特定交易日 (TDAYS)	22
3.11.2	日期偏移函数 (TDAYSOFFSET)	23
3.11.3	交易日统计 (TDAYSCOUNT)	23
3.12	日期宏	24
3.12.1	通用日期宏	24
3.12.2	特殊日期宏	25
4	WINDR 应用案例	26
4.1	提取数据	26
4.1.1	提取历史交易报价	26
4.1.2	提取分钟序列数据	26
4.1.3	提取盘口买卖盘数据	27
4.1.4	提取截面数据	27
4.1.5	提取实时行情数据	27
4.1.6	提取财务数据	28
4.1.7	提取债券估值数据	28
4.1.8	提取数据集	28
4.1.9	提取资管报表数据	30
4.1.10	提取交易日期	30
4.2	读取股票日 K 线价格并绘制价格图	30

4.3 DEMO 程序介绍	31
4.3.1 wsd_quant_demo	31
4.3.2 wsi_demo	32
4.3.3 wst_demo	33
4.3.4 wsq_demo	34
5 常见问题	38
5.1 安装及注册	38
5.2 读取指标数据	38
5.3 交易接口查询返回的数据字段	41
5.3.1 资金查询返回消息	41
5.3.2 持仓查询返回消息	43
5.3.3 当日委托查询返回消息	44
5.3.4 当日成交查询返回消息	46
5.3.5 营业部查询返回消息	47
5.3.6 股东查询返回消息	48
5.3.7 券商（期货商）信息返回	48
5.3.8 已登录账户信息返回	48

1 WindR 接口说明

1.1 WindR 接口概述

大数据时代已经来临！为满足我们用户在构建模型，量化研究中对大数据量的渴求，Wind 资讯将陆续推出一整套数据接口。

2012 年 8 月，我们在 Excel 中推出了一系列 WX 多值函数，数据范围涵盖基本面序列数据，日间与日内高频行情数据，历史快照与实时截面数据，日内分钟更新数据等。

2012 年 12 月，我们推出 Matlab 数据接口 Beta 版本，方便用户远程访问 Wind 资讯云数据服务，快速提取各类行情与基本面数据。

2013 年 4 月，我们推出了 3000 多个基本面及行情指标，量化功能大大加强。

2013 年 6 月，我们推出 R 数据接口 Beta 版本，在支持多种量化研究工具方面又有所提升，用户可以借助强大的 R 软件包，实现各种金融建模需求。

2013 年 8 月，WindR 接口增加了交易接口和条件选股功能，可提取的指标数量也进一步增加。

1.2 WindR 接口安装

1.2.1 WindR 对系统环境要求

- Windows 系统，支持 32 位和 64 位系统；
- R2.15.0 以上的 R 环境，包括 R2.15.X, R3.X.X 等等；
- Wind 终端最新版 2013 年 5 月 28 日后版本；

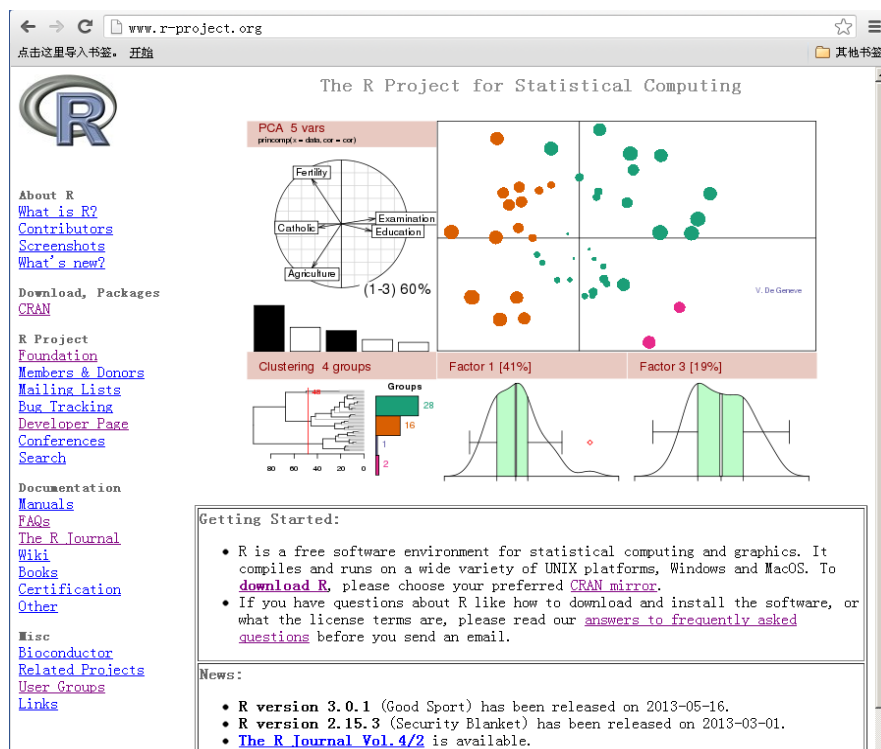
- 安装时由于需要写注册表，因此需要系统管理员权限。

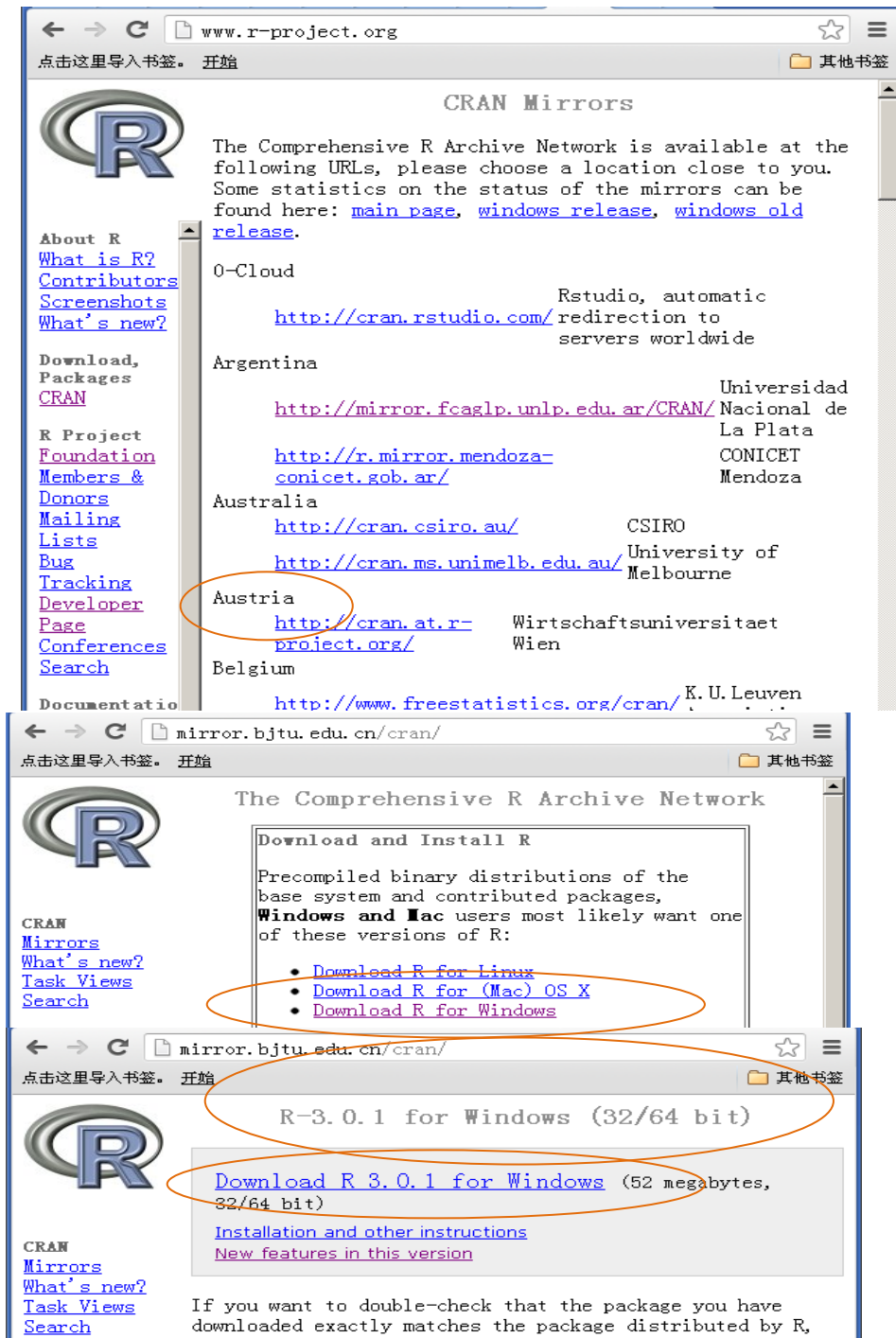
1.2.2 R 环境安装

R 是一个有着统计分析功能及强大作图功能，在 GNU 协议下免费发行的软件，与 Matlab 相比，R 更擅长统计分析，具有更好的开放性，在金融和统计领域具有很强的应用前景。

R 官方下载地址为 <http://www.r-project.org/>。进入到该界面后，点击 download R 链接，会出来 CRAN Mirrors 界面，用户可以选择一个离自己较近的站点，点击后，就进入到下载界面。具体请看以下界面。

下载后，直接运行即可。运行时需要写注册表，因此最好拥有系统管理员权限，否则可能在安装 WindR 插件时需要手工安装。



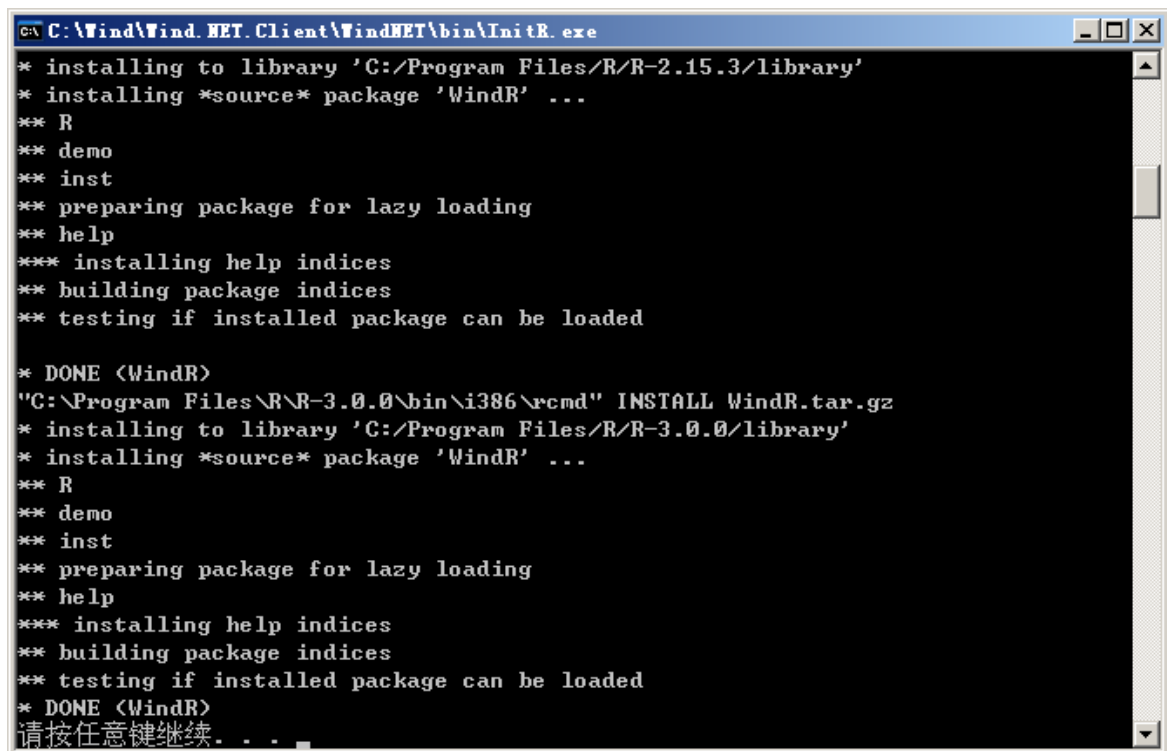


1.2.3 正常 WindR 接口安装

- 1) 确保达到 1.2.1 中的安装要求，并确保关闭 R 环境，以及用到控件的 Matlab 程序和 c++ 环境等；
- 2) 打开 Wind 资讯终端，点击“量化”选项，出现下方的界面，点击“R 插件”，会弹出广告说明；



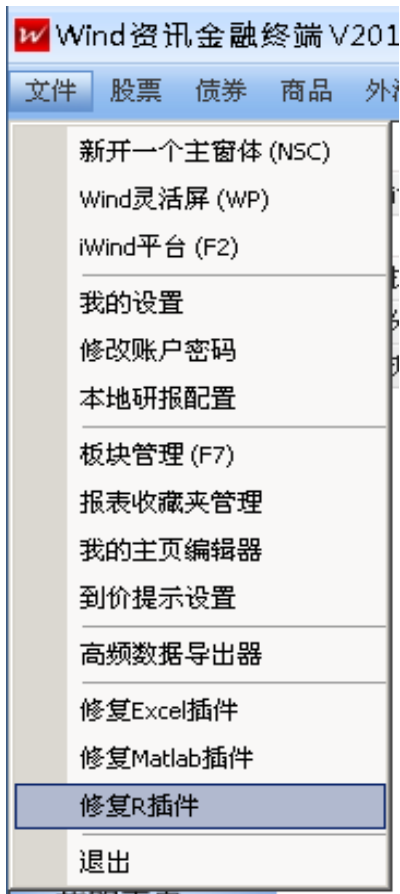
- 3) 用户可以在“文件”菜单下选择“修复 R 插件”，或者输入“RepairR”命令，会出现下面的界面：



```
C:\Wind\Wind.NET.Client\WindNET\bin\InitR.exe
* installing to library 'C:/Program Files/R/R-2.15.3/library'
* installing *source* package 'WindR' ...
** R
** demo
** inst
** preparing package for lazy loading
** help
*** installing help indices
** building package indices
** testing if installed package can be loaded

* DONE <WindR>
"C:\Program Files\R\R-3.0.0\bin\i386\rcmd" INSTALL WindR.tar.gz
* installing to library 'C:/Program Files/R/R-3.0.0/library'
* installing *source* package 'WindR' ...
** R
** demo
** inst
** preparing package for lazy loading
** help
*** installing help indices
** building package indices
** testing if installed package can be loaded

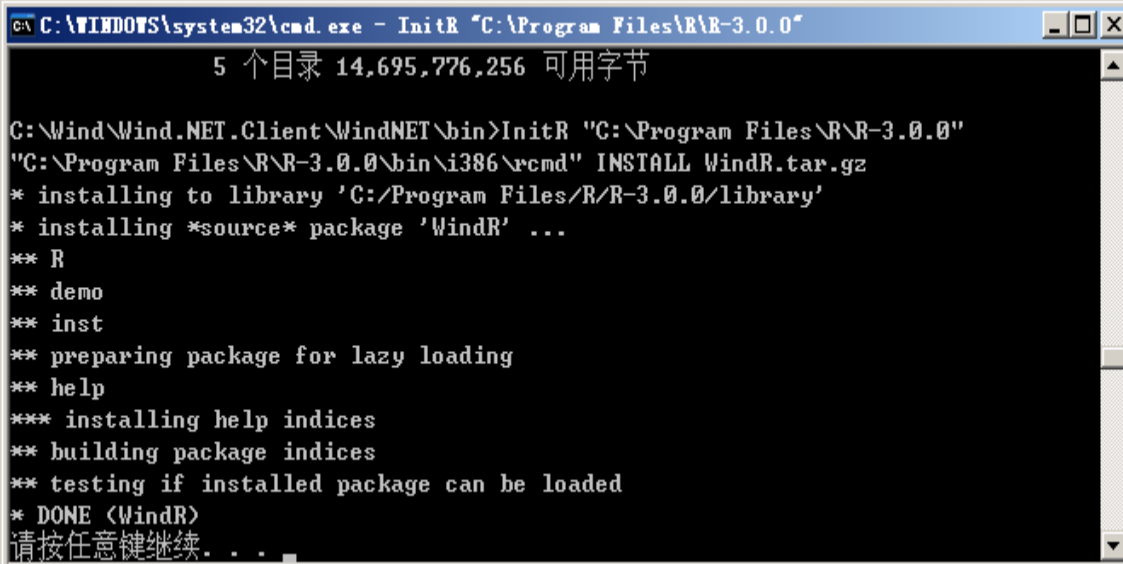
* DONE <WindR>
请按任意键继续. . .
```



4) 按任意键 WindR 安装过程结束。

1.2.4 特殊安装 WindR 方式

- 1) 确保达到 1.2.1 中的安装要求，并确保关闭 R 环境。
- 2) 通过 Windows 的 cmd 命令，进入到 Wind 终端安装目录中，一般在 C:\Wind\Wind.NET.Client\WindNET\bin; 输入 InitR "R 安装的目录"，如下图，图中的"C:\Program Files\R\R-3.0.0"为用户 R 语言的安装目录，请注意使用引号，并且最后没有"\":



```
C:\WINDOWS\system32\cmd.exe - InitR "C:\Program Files\R\R-3.0.0"
5 个目录 14,695,776,256 可用字节

C:\Wind\Wind.NET.Client\WindNET\bin>InitR "C:\Program Files\R\R-3.0.0"
"C:\Program Files\R\R-3.0.0\bin\i386\rcmd" INSTALL WindR.tar.gz
* installing to library 'C:/Program Files/R/R-3.0.0/library'
* installing *source* package 'WindR' ...
** R
** demo
** inst
** preparing package for lazy loading
** help
*** installing help indices
** building package indices
** testing if installed package can be loaded
* DONE (WindR)
请按任意键继续...
```

- 3) 按任意键 WindR 安装过程结束。

另：也可在 R 中运行：

```
install.packages("C:/Wind/Wind.NET.Client/WindNET/bin/
WindR.tar.gz", repos = NULL, type="source"); 注意根据实际情况修改其中的路径。
```

1.3 WindR 接口向导界面

用户可以用向导来熟悉 Wind R 数据接口的各项功能，可以生成提取数据的命令行或者直接提取数据到 R 变量当中。

在 R 命令窗口下键入如下命令。

```
>library(WindR)
>w.start()
```

就会在 R 窗口上弹出向导。



用户可以点击不同选项执行不同操作。R 提取数据的功能都可以通过向导实现。

用户可以随时通过 `w.menu()` 隐藏或开启该导航界面

1.4 WindR 获取帮助途径

用户可以通过如下方式获取 WindR 帮助。

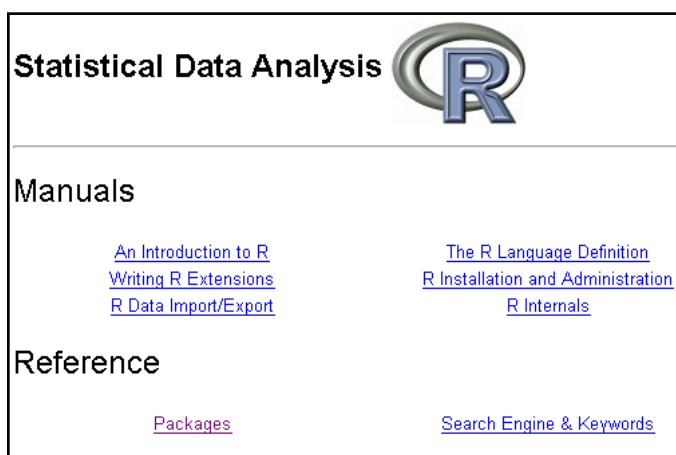
1.4.1 本用户手册

本用户手册会介绍产品的功能和用户经常反馈的信息。查看本手册是一个不错的学习途径。

对于第一次使用者来说，请从本手册开始浏览，这样不会错过一些重要的信息。

1.4.2 R 里面的帮助文档

- 1) 在 R 主帮助文档中，点击“Packages”，列出所有的安装包，点击其中的“WindR”然后就出现 WindR 帮助文档。



stats	The R Stats Package
stats4	Statistical Functions using S4 Classes
survival	Survival Analysis
tcltk	Tcl/Tk Interface
tools	Tools for Package Development
TTR	Technical Trading Rules
utils	The R Utils Package
WindR	Financial Data API of Shanghai Wind Information Co., Ltd
xts	eXtensible Time Series
zoo	S3 Infrastructure for Regular and Irregular Time Series (Z's ordered observations)

Financial Data API of Shanghai Wind Information Co., Ltd



Documentation for package 'WindR' version 1.00

- [DESCRIPTION file](#).
- [Code demos](#). Use [demo\(\)](#) to run them.

Help Pages

WindR-package	Financial Data API of Wind Information
w.asDateTime	Time function
w.callback	Callback function used in API
w.cancelRequest	Cancel a request
w.clearAllRequest	Cancel All Request
w.close	Stop Wind Data API
w.demoCallback	Demo callback function
w.isconnected	To check whether WindR started OK.
w.menu	Toggle the toolbar; open the guide dialog.
w.start	Start Wind Data API, login to Wind Server.
w.stop	Stop Wind Data API
w.tdays	Retrieve valid days between two days.
w.tdayscount	Retrieve duration between two days.
w.tdaysoffset	Retrieve a day based on the input day
w.wpf	Retrieve portfolio Information.
w.wsd	Retrieve Wind daily data
w.wset	Retrieve some base info about market.
w.wsi	Retrieve Wind inter-day data
w.wsq	Retrieve Wind quote data
w.wss	Retrieve Wind snapshot data
w.wst	Retrieve Wind tick data
WindR	Financial Data API of Wind Information

在该帮助界面中，用户可以分别点击每个链接得到每个函数的说明。特别的可以点击 demo 链接获得 demo 程序源代码。

2) 使用?调出帮助文档

用户在装载 WindR 包后,即使用 `library(WindR)` 后, 可以使用

```
>?WindR
```

```
>?w.wsd
```

```
>??wsd
```

等等查看各种命令的帮助;

也可以使用

```
> demo(package='WindR')
```

```
>demo(wsq_demo)
```

查看 demo 程序运行效果。

具体 demo 程序可以使用如下命令查看


```
>??wsi_demo
```

1.4.3 量化交易群和 R 语言交流群

用户可以通过 WM 申请加入中国量化交易群 (群号 59289) 和 R 交流群 (群号 60747)，在这两个群中学习 WindR 接口使用和量化交易知识。

1.5 WindR 接口相关规范

1.5.1 命令区分大小写，且“w.”不能省略

如：`w.tdaysoffset(-1)` 不能写成 `tdaysoffset(-1)`，或者 `w.TDaysOffset(-1)`；

1.5.2 单字节码和双字节码的问题

中文常使用双字节编码，这在 R 中使用时就会错误。比如引号、逗号、括号等；

1.5.3 品种、指标、参数等引号内的部分不区分大小写

比如 `w.wsd('090007.IB','close',Sys.Date()-5,Sys.Date(),'Priceadj=F;tradingcalendar=NIB')`
和 `w.wsd('090007.ib','CLOSE',Sys.Date()-5,Sys.Date(),'Priceadj=f;tradingcalendar=nib')` 一样；

1.5.4 参数支持数组输入

比如 `w.wss("600000.SH,600005.SH,600004.SH,600007.SH","roe_avg,roa","rptDate=20121231")`
可以写成：

```
code<-c('600000.SH','600005.SH','600004.SH')
field<-c('roe_avg','roa')
w.wss(code,field,"rptDate=20121231")
```

1.5.5 时间、日期支持 R 语言的时间、日期格式

比如 `w.wsd('600000.SH', 'open', '20130505')`
也可以写成 `w.wsd('600000.SH', 'close', Sys.Date()-10);`

1.5.6 参数中有缺省值的可以不用输入

比如 `w.wsd` 定义为 `w.wsd(codes, fields, beginTime, endTime = Sys.time(), options = "")`，可选参数和结束时间都有缺省值，因此用户可以不输入可选参数，也可以不输入结束时间。不输入时使用缺省值。

如：`w.wsd('600000.SH', 'open', '20130505')` 等同于 `w.wsd('600000.SH', 'open', '20130505', Sys.time());`

1.5.7 可以带参数名输入

比如 `w.wsd` 定义为 `w.wsd(codes, fields, beginTime, endTime = Sys.time(), options = "")`。

`w.wsd("600000.SH", "high", "2013-05-09", Sys.Date(), "Period=W")` 等同于
`w.wsd("600000.SH", "high", "2013-05-09", Sys.Date(), options="Period=W")` 等同于
`w.wsd("600000.SH", "high", "2013-05-09", options="Period=W", endTime=Sys.Date())`

带参数名输入后，参数顺序就可以变化；

1.5.8 Showblank 参数

Showblank 参数可以指定对返回的 NaN 单元进行特别处理，如：

把 NaN 用 -1 替换:

```
w.wsd('600001.sh', 'open,close', '20130707', '20130909', 'showblank=-1');  
或 w.wsd('600001.sh', 'open,close', '20130707', '20130909', showblank=-1);
```

把 NaN 用 0 替换:

```
w.wsd('600001.sh', 'open,close', '20130707', '20130909', 'showblank=0');  
或 w.wsd('600001.sh', 'open,close', '20130707', '20130909', showblank=0);
```

1.5.9 交易接口中 Showfields 参数

交易接口返回的内容的指标根据具体情况会有变化，而有的情况下，客户需要指定确切的返回字段和顺序，此时可以使用 showfields 参数。如：

```
w.tquery(1, logonid=1, 'showfields=securitycode, Profit, securityBalance')  
或: w.tquery(1, logonid=1, showfields='securitycode, Profit, securityBalance ')
```

1.5.10 ErrorCode 定义

ErrorCode=0 表示操作成功。

其他:

-40520001	未知错误	-40520002	内部错误
-----------	------	-----------	------

-40520003	系统错误	-40520004	登录失败	
-40520005	无权限	-40520006	用户取消	
-40520007	无数据	-40520008	超时错误	-40521010 超时错误
-40520009	本地 WBOX 错误	-40520010	需要内容不存在	
-40520011	需要服务器不存在	-40520012	引用不存在	
-40520013	其他地方登录错误	-40520014	未登录使用 WIM 工具，故无法登录	
-40520015	连续登录失败次数过多			
-40521001	IO 操作错误	-40521002	后台服务器不可用	
-40521003	网络连接失败	-40521004	请求发送失败	
-40521005	数据接收失败	-40521006	网络错误	
-40521007	服务器拒绝请求	-40521008	错误的应答	
-40521009	数据解码失败	-40521010	网络超时	
-40521011	频繁访问			

-40522001	无合法会话	-40522002	非法数据服务
-40522003	非法请求	-40522004	万得代码语法错误
-40522005	不支持的万得代码	-40522006	指标语法错误
-40522007	不支持的指标	-40522008	指标参数语法错误
-40522009	不支持的指标参数	-40522010	日期与时间语法错误
-40522011	不支持的日期与时间	-40522012	不支持的请求参数
-40522013	数组下标越界	-40522014	重复的 WQID
-40522015	请求无相应权限	-40522016	不支持的数据类型
-40522017	数据提取量超限		

2 Wind R 插件命令说明

2.1 library(WindR): 装载 WindR 包

在具体运行各种命令前，用户首先应装载 WindR 包，即输入 `library(WindR)`，也可以 `require(WindR)`。

实例: `library(WindR)`

2.2 ?WindR: 启动 WindR 帮助文档

装载 WindR 后，用户通过 `?WindR`, `?w.wsd` 等获得各种帮助

实例: `library(WindR)`
`?WindR`
`?w.wsd`

2.3 w.start: 启动 WindR

在真正开始操作之前，可以使用该命令登录并启动 windR 插件。用户可以使用 `?w.start` 查看命令说明。

实例: `library(WindR)`
`w.start()`; #缺省显示导航界面，命令超时时间为 300 秒
`w.start(showmenu=FALSE)`; #不显示登录界面，命令超时时间为 300 秒
`w.start(waitTime=60,showmenu=FALSE)`; #不显示登录界面，命令超时时间设置成 60 秒

注:

w.start 不重复启动，若需要改变参数，如超时时间，用户可以使用 w.stop 命令先停止后再启动。

2.4 w.stop: 停止 WindR

当需要停止 WindR 时，可以使用该命令。用户可以使用 ?w.stop 查看命令说明。

实例：`library(WindR)`
`w.start();`
`w.stop()`

注：退出时，会自动执行 `w.stop()`，用户一般并不需要执行 `w.stop`。

2.5 w.menu: 显示导航界面

当需要显示导航界面时，可以使用该命令。用户可以使用 ?w.menu 查看命令说明。

实例：`library(WindR)`
`w.start(showmenu=F);` #不显示导航界面
`w.menu()` #显示导航界面
`w.menu()` #第二次运行关闭显示导航界面
`w.menu('wsd');` #显示 wsd 导航界面
`w.menu('wset');` #显示 wset 导航界面

以下为导航总界面，用户可以点击各按钮使用具体功能点。



注：目前支持 wsd, wss, wsi, wst, wsq, wset, weqs, wpf, tdays, tdayscount, tdaysoffset, trade 等参数。

2.6 w.isconnected: 判断是否已经登录

可以使用该命令确定 windR 是否登陆成功。用户可以使用 `?w.isconnected` 查看命令说明。

实例：`library(WindR)`
`w.start(showmenu=F);` #不显示导航界面
`w.isconnected()` #即判断 WindR 是否已经登陆成功

2.7 w.cancelRequest: 取消订阅

该命令用来根据订阅请求的 id，取消订阅（目前只有 w.wsq 订阅）。用户可以使用 `?w.cancelRequest` 查看命令说明。

实例：`library(WindR)`
`w.start(showmenu=F);` #不显示导航界面
`data<-w.wsq("600000.SH","rt_low,rt_last_vol",func=w.demoCallback);` #订阅
#等待回调，用户可以根据实际情况写回调函数
#....
`w.cancelRequest(data$RequestID);` #根据刚才 wsq 返回的请求 ID，取消订阅

注：可以象 `w.cancelRequest(3)` 一样，输入一个 id 的数字，而取消某订阅

2.8 w.asDateTime: 把数字化时间格式转换成 R 语言时间格式

为了用户的使用方便,接口函数中时间指标是以距离某个时间点的天数形式输出的,是一个浮点数,而 `asDateTime` 是用来把这种格式的时间值转换成 R 语言时间格式。

实例: `library(WindR)`

```
w.start(showmenu=F);#不显示导航界面
```

```
data<-w.wss("600000.SH,000002.SZ","last_trade_day,pre_close")#取两支股票最近交易日,以及前收盘价
```

```
#返回的最近交易日是数字化时间格式,用 w.asDateTime 转换,假设当前是 2013.7.4 号
```

```
w.asDateTime(data$Data[,2]) #转换成时间格式,结果为"2013-07-04 00:00:00 GMT" "2013-07-04 00:00:00 GMT"
```

```
w.asDateTime(data$Data[,2],asdate=T) #转换成日期形式,结果为"2013-07-04" "2013-07-04"
```

2.9 w.wsd: 获取历史序列数据

该命令用来获取选定证券品种的历史序列数据,包括日间的行情数据、基本面数据以及技术数据指标。用户可以使用 `?w.wsd` 查看命令说明。命令原型为: `data<- w.wsd(品种代码,指标,开始日期,结束日期,可选参数);`

返回参数为:

`data$Data` 返回的序列数据,为 `data.frame` 格式;

`data$Code` 数据对应的 WindCode 代码;

`data$ErrorCode` 命令是否成功的错误码,0 表示成功。

实例: `library(WindR)`

```
w.start(showmenu=F);#不显示导航界面
```

```
data<-w.wsd("600000.SH","close,amt","2013-04-30",Sys.Date()-1)#取浦发银行收盘价等信息
```

```
data<-w.wsd("600000.SH","close,amt",Sys.Date()-100) #取浦发银行收盘价等信息
```

- 注：1) 一次只能一个品种，并且品种名带有“.SH”等后缀；
2) 指标和可选参数也可以用数组实现；
3) 日期支持 R 中时间和日期格式；
4) 可选参数有很多种；
5) 用户可以用 `w.menu('wsd')` 显示导航界面，帮助创建命令。

2.10 w.wsi: 获取分钟数据

该命令用来获取选定证券品种的分钟 K 线数据，包含历史和当天，分钟周期可以指定，技术指标参数可以自定义设置。用户可以使用 `?w.wsi` 查看命令说明。命令原型为：`data<- w.wsi(品种代码,指标,开始时间,结束时间,可选参数)`；

返回参数为：

`data$Data` 返回的序列数据，为 `data.frame` 格式；
`data$Code` 数据对应的 WindCode 代码；
`data$ErrorCode` 命令是否成功的错误码，0 表示成功。

实例：`library(WindR)`

```
w.start(showmenu=F); #不显示导航界面  
data<-w.wsi("600000.SH","close,amt","2013-05-30 9:00:00") #取浦发银行分钟收盘价等信息  
data<-w.wsi("600000.SH","close,amt",Sys.Date()-10, Sys.time()) #取浦发银行分钟收盘价等信息
```

- 注：1) 一次只能一个品种，并且品种名带有“.SH”等后缀；
2) 指标和可选参数也可以用数组实现；

- 3) 日期支持 R 中时间和日期格式;
- 4) 可选参数有很多种;
- 5) 用户可以用 `w.menu('wsi')` 显示导航界面, 帮助创建命令;
- 6) 一次只能取 3 个月内数据。

2.11 w.wst: 获取日内 tick 级别数据

该命令用来获取选定证券品种的日内盘口买卖十档快照数据和分时成交数据 (tick 数据)。用户可以使用 `?w.wst` 查看命令说明。

命令原型为: `data<- w.wst (品种代码, 指标, 开始时间, 结束时间, 可选参数);`

返回参数为:

- `data$Data` 返回的序列数据, 为 `data.frame` 格式;
- `data$Code` 数据对应的WindCode代码;
- `data$ErrorCode` 命令是否成功的错误码, 0表示成功。

实例:

```
library(WindR)
w.start(showmenu=F);#不显示导航界面
data<-w.wst("600000.SH","open",Sys.time()-2*3600,Sys.time())#取浦发银行 tick 数据信息
```

- 注:
- 1) 一次只能一个品种, 并且品种名带有“.SH”等后缀;
 - 2) 指标和可选参数也可以用数组实现;
 - 3) 日期支持 R 中时间和日期格式;
 - 4) 可选参数有很多种;
 - 5) 用户可以用 `w.menu('wst')` 显示导航界面, 帮助创建命令;

6) 目前只支持当天数据（假日可以取上一交易日数据），并且只有时间有意义，日期无意义。

2.12 w.wss: 获历史截面数据

命令用来获取选定证券品种的历史截面数据，比如取沪深 300 只股票的 2012 年 3 季度的净利润财务指标数据。 用户可以使

用 `?w.wss` 查看命令说明。命令原型为：`data<- w.wss(品种代码,指标,可选参数)`；

返回参数为：

`data$Data` 返回的序列数据，为 `data.frame` 格式；
`data$Time` 数据对应的时间信息。
`data$ErrorCode` 命令是否成功的错误码，0 表示成功。

实例：

```
library(WindR)
w.start(showmenu=F);#不显示导航界面
data<-w.wss("600000.SH,000001.SZ","eps_ttm,orps,surpluscapitalps","rptDate=20121231")
#取浦发银行等财务数据信息
```

注：1) 一次只能取一个报告期，但可以取多个品种数据
2) 品种代码、指标和可选参数也可以用数组实现；
3) 可选参数有很多种；
4) 用户可以用 `w.menu('wss')` 显示导航界面，帮助创建命令；

2.13 w.wsq: 获取和订阅实时行情数据

命令用来获取选定证券品种的当天实时指标数据，数据可以一次性请求，也可以通过订阅的方式获取。用户可以使用 `?w.wsq` 查看命令说明。命令原型为：`data<- w.wsq(品种代码, 指标, 可选参数, 回调函数)`；

对于一次性数据获取时（也即没有设置回调函数），返回参数为：

`data$Data` 返回的序列数据，为 `data.frame` 格式；

`data$Time` 数据对应的时间信息。

`data$ErrorCode` 命令是否成功的错误码，0表示成功。

对于订阅，返回参数为：

`data$RequestID` 返回订阅 ID，稍后可以使用 `w.cancelRequest(data$RequestID)` 取消订阅。

`data$ErrorCode` 命令是否成功的错误码，0表示成功。

实例：

```
library(WindR)
w.start(showmenu=F);#不显示导航界面
w.wsq("600000.SH,000001.SZ","rt_last,rt_last_vol")#取浦发银行等股票当前行情信息
data<-w.wsq("600000.SH","rt_low,rt_last_vol",func=w.demoCallback)#订阅浦发银行等股票当前行情信息
#.....
w.cancelRequest(data$RequestID)#取消订阅
```

注：

- 1) 用户自己定义的回调函数格式请参考 `w.demoCallback`，回调函数中不应处理复杂的操作。并且用户可以使用 `?w.demoCallback` 看帮助文档；
- 2) 品种代码、指标和可选参数也可以用数组实现；用户可以一次提取或者订阅多个品种数据
- 3) 用户可以用 `w.menu('wsq')` 显示导航界面，帮助创建命令；

4) 订阅时, API 发现用户订阅内容发生变化则调用回调函数, 并且只把变动的内容传递给回调函数。

2.14 w.wset: 获取板块、指数等成分数据

命令用来获取数据集信息, 包括板块成分、指数成分、ETF 申赎成分信息、分级基金明细、融资标的、融券标的、融资融券担保品、回购担保品、停牌股票、复牌股票、分红送转。参数设置为起止日期、板块名称等。用户可以使用 `?w.wset` 查看命令说明。命令原型为: `data<- w.wset(数据集名称,可选参数);`

返回参数为:

- `data$Data` 返回的序列数据, 为 `data.frame` 格式;
- `data$Time` 数据对应的时间信息。
- `data$ErrorCode` 命令是否成功的错误码, 0表示成功。

实例:

```
library(WindR)
w.start(showmenu=F); #不显示导航界面
data<-w.wset("SectorConstituent","date=20130608;sector=全部 A 股") #取全部 A 股股票代码、名称信息
#取沪深 300 指数中股票代码和权重
data<-w.wset("IndexConstituent","date=20130608;windcode=000300.SH;field=wind_code,i_weight")
#取停牌信息
w.wset("TradeSuspend","startdate=20130508;enddate=20130608;field=wind_code,sec_name,suspend_type,suspend_reason")
#取 ST 股票等风险警示股票信息
data<-w.wset("SectorConstituent","date=20130608;sector=风险警示股票;field=wind_code,sec_name")
```

注:

- 1) 可选参数也可以用数组实现;
- 2) 用户可以用 `w.menu('wset')` 显示导航界面, 帮助创建命令;

2.15 w.weqs: 获取条件选股结果

用来读取某个条件选股的结果。用户可以使用 `?w.weqs` 查看命令说明。命令原型为: `data<- w.weqs(filtername,...)`;

返回参数为:

`data$Data` 返回的序列数据, 为 `data.frame` 格式;
`data$Time` 数据对应的时间信息。
`data$ErrorCode` 命令是否成功的错误码, 0表示成功。

实例:

```
library(WindR)
w.start(showmenu=F); #不显示导航界面

#事先已经创建了“七日新低”这个条件选股。(可以在终端上输入 eqs 创建)
w.weqs('七日新低')
```

注:

- 1) 可选参数也可以用数组实现;
- 2) 用户可以用 `w.menu('weqs')` 显示导航界面, 帮助创建命令;

2.16 w.wpf: 获取资产管理、组合管理数据

用来读取交易账户与资管账户中的报表数据。用户可以使用 `?w.wpf` 查看命令说明。命令原型为: `data<- w.wpf(产品名, 数据`

表名, 可选参数);

返回参数为:

data\$Data 返回的序列数据, 为 data.frame 格式;
data\$Time 数据对应的时间信息。
data\$ErrorCode 命令是否成功的错误码, 0表示成功。

实例:

```
library(WindR)
```

```
w.start(showmenu=F); #不显示导航界面
```

```
#返回组合管理演示 产品中的 组合日结算数据
```

```
data<-w.wpf("组合管理演示", "PMS.PortfolioDaily",  
"startdate=20130509;enddate=20130609;reportcurrency=CNY;owner=")
```

```
data<-w.wpf("总账-MMM", "AMS.PortfolioDailySerial") #取资产管理 AMS 中"总账-MMM"产品日数据序列信息
```

注:

- 1) 可选参数也可以用数组实现;
- 2) 用户可以用 w.menu('wpf'), w.menu('ams'), w.menu('pms') 显示导航界面, 帮助创建命令;
- 3) ams 需要先授权, 并创建了产品之后才能使用, 具体可以联系 Wind 客服;
- 4) pms 有缺省产品“组合管理演示”, 用户可以使用。

2.17 交易相关函数

2.17.1 w.tlogon 交易登录

命令用来登录交易系统。用户可以使用 `?w.tlogon` 查看命令说明。命令原型为：`data <- w.tlogon(BrokerID, DepartmentID, LogonAccount, Password, AccountType, ...)`

返回参数为：

`data$Data` 返回的序列数据，为 `data.frame` 格式；
`data$ErrorCode` 命令是否成功的错误码，0表示成功。

例如：

```
> w.tlogon('0000','0','000100000090',c('aaa',aaa',3),'SH')
#两个密码是对的，一个密码是错的。
[1] ""
$Data
      LogonID      LogonAccount      AccountType      ErrorCode
ErrorMsg
1          0 000100000090          SZSHA          0
OK
2          1 000100000090          SZSHA          0
OK
3      NA 000100000090          SZSHA      2 200 登录失败:资金账户
000100000090 密码错误!

$ErrorCode
[1] 2
```

注：

- 1) 本命令支持向量操作，也即每个参数都可以使用数组输入，对于只有一个元素的参数会自动扩充；
- 2) 数字和字符串具有同等效果
- 3) 用户可以用 `w.menu('tlogon')` 显示导航界面，帮助创建命令；
- 4) 有 WFT 账号的用户，已经自动开通模拟账号，其中股票模拟账号为：WFT 账号+01，期货为 WFT 账号+02

2.17.2 w.tlogout 交易登出

命令用来登出交易系统。用户可以使用 `?w.tlogout` 查看命令说明。命令原型为：`data <- w.tlogout((LogonID = ""))`

返回参数为：

data\$Data 返回的序列数据，为 data.frame 格式；
data\$ErrorCode 命令是否成功的错误码，0表示成功。

例如：

```
> w.tlogout(c(0,1))
$Data
  LogonID ErrorCode ErrorMsg
1       0         0  logout
2       1         0  logout

$ErrorCode
[1] 0
```

注：

- 1) 本命令支持向量操作，也即每个参数都可以使用数组输入，对于只有一个元素的参数会自动扩充；
- 2) 数字和字符串具有同等效果
- 3) 只有一个交易登录时，可以不输入 LogonID。
- 4) 用户可以用 w.menu('tlogout') 显示导航界面，帮助创建命令；

2.17.3 w.torder 委托下单

命令用来委托下单。用户可以使用 ?w.torder 查看命令说明。命令原型为：
data <-w.torder(SecurityCode, TradeSide, OrderPrice, OrderVolume, ..., MarketType = "", OrderType = "", HedgeType = "", LogonID = "")

返回参数为：

data\$Data 返回的序列数据，为 data.frame 格式；
data\$ErrorCode 命令是否成功的错误码，0表示成功。

例如：w.torder('600000.SH', 'buy', 9.8, 100)

例如：

```
> code=c('600000.SH','000001.SZ')
> price=w.wsq(code,'rt_last')$Data$RT_LAST
> w.torder(code,'buy',price,100,logonid=0)
$Data
  RequestID SecurityCode TradeSide OrderPrice OrderVolume
LogonID ErrorCode ErrorMsg
1         6  600000.SH         1      8.31         100      0
0 Sending ...
2         7  000001.SZ         1     10.43         100      0
0 Sending ...

$ErrorCode
[1] 0
```

注:

1) 本命令支持向量操作, 也即每个参数都可以使用数组输入, 对于只有一个元素的参数会自动扩充;

2) 数字和字符串具有同等效果

3) 只有一个交易登录时, 可以不输入 LogonID, 否则一定需要输入, 即用 LogonID=xxxx 方式输入。

4) 用户可以用 w.menu('torder') 显示导航界面, 帮助创建命令;

5) TradeSide 可以为: 1/buy; 2/short; 3/cover; 4/sell; 5/coverToday; 6/sellToday

6) OrderType 可以为: 0/LMT; 1/BOC; 2/BOP; 3/ITC; 4/B5TC; 5/FOK; 6/B5TL;

7) 当用户输入的代码没有带. 的市场后缀时, 需要提供 MarketType, MarketType 可以取: 0/SZ; 1/SZ; 2/OC; 6/HK; 7/CZC; 8/SHF;

9/DCE; 10/CFE;

8) 可以通过 w.tquery('order', requestid=xxx) 查询委托情况

9) 期货套保账号时一定要加上 HedgeType=HEDG/1, 因为缺省是投机 SPEC
0

2.17.4 w.tcancel 撤销委托

命令用来撤销委托。用户可以使用 ?w.tcancel 查看命令说明。命令原型为:
data <-w.tcancel(OrderNumber, ..., MarketType = "", LogonID = "")

返回参数为:

data\$Data 返回的序列数据, 为 data.frame 格式;

data\$ErrorCode 命令是否成功的错误码, 0表示成功。

例如:

```
> w.tcancel(c(1231,1232),logonid=0)
```

```
$Data
```

	OrderNumber	LogonID	ErrorCode	ErrorMsg
1	1231	0	0	Sending ...
2	1232	0	0	Sending ...

```
$ErrorCode
```

```
[1] 0
```

注:

1) 本命令支持向量操作, 也即每个参数都可以使用数组输入, 对于只有一个元素的参数会自动扩充;

2) 数字和字符串具有同等效果

3) 只有一个交易登录时, 可以不输入 LogonID, 否则一定需要输入, 即用 LogonID=xxxx 方式输入。

- 4) 用户可以用 `w.menu('tcancel')` 显示导航界面, 帮助创建命令;
- 5) 当用户有很多笔不同市场的下单时, `OrderNumber` 可能会有重复, 此时需要使用 `MarketType` 区别, `MarketType` 可以取: 0/SZ; 1/SZ; 2/OC; 6/HK; 7/CZC; 8/SHF; 9/DCE; 10/CFE;

2.17.5 w.tquery 交易查询

命令用来查询交易相关各信息。用户可以使用 `?w.tquery` 查看命令说明。命令原型为: `data <- w.tquery(qrycode, ..., LogonID = "", RequestID = "", OrderNumber = "", SecurityCode = "", options = "")`

返回参数为:

`data$Data` 返回的序列数据, 为 `data.frame` 格式;
`data$ErrorCode` 命令是否成功的错误码, 0表示成功。

查询返回的内容很多, 请参考常见问题, 查看返回的各字段意义。

例如:

```
> w.tquery(0, logonid=c(0,1)) #查询资金情况
$Data
  MoneyType AvailableFund BalanceFund SecurityValue FundAsset
TotalAsset Profit FundFrozen OtherFund BuyFund SellFund
Remark DepartmentID Customer AssetAccount
1 CNY 7846410 9130470000 9992120000 7863660
9999990000 -11358.5 17246.5 0 9122950000 345126 新开户
0001 000000000090 000100000090
2 CNY 7846410 9130470000 9992120000 7863660
9999990000 -11358.5 17246.5 0 9122950000 345126 新开户
0001 000000000090 000100000090
LogonID ErrorCode ErrorMessage
1 0 0 OK
2 1 0 OK

$ErrorCode
[1] 0

> w.tquery(2, logonid=0) #查询委托情况
> vv$Data[1230:1231,]
  OrderNumber OrderStatus SecurityCode SecurityName
TradeSide OrderPrice OrderVolume OrderTime
1 1231 Normal 600000.SH 浦发银行 Buy
8.31 100 41500.77
2 1232 Normal 000001.SZ 平安银行 Buy
10.43 100 41500.77
```

```

TradedPrice TradedVolume CancelVolume LastPrice MadeAmt
OrderFrozenFund MoneyType Remark LogonID
1          8.14          100          0          0          814
837          0  已成          0
2          10.22          100          0          0          1022
1048          0  已成          0
      ErrorCode ErrorMessage
1          0          OK
2          0          OK

```

注:

- 1) 除 qrycode 外, 本命令支持向量操作, 也即其他每个参数都可以使用数组输入, 对于只有一个元素的参数会自动扩充;
- 2) 数字和字符串具有同等效果
- 3) 只有一个交易登录时, 可以不输入 LogonID, 否则一定需要输入, 即用 LogonID=xxxx 方式输入。
- 4) 用户可以用 w.menu('tquery') 显示导航界面, 帮助创建命令;
- 5) qrycode 可取: 0/capital 资金查询; 1/position 持仓查询; 2/order 今日委托查询; 3/trade 今日成交查询; 4/departement 营业部查询; 5/account 股东账号查询; 6/broker 经济商查询; 7/logonid 登录的账号查询
- 6) 今日委托查询 2/order 时可以依据委托 order 返回的 requestid 查询, 该查询立即返回, 返回服务器已经返回的信息;
- 7) 营业部查询时 4/departement, 需要输入 brokerid 参数

2.18 w.tdays, w.tdaysoffset, w.tdayscount: 日期函数

2.18.1 w.tdays: 返回区间内的日期序列

命令用来获取两个时间区间内的某种规则下的日期序列。用户可以使用 ?w.tdays 查看命令说明。命令原型为: data<- w.tdays(开始时间, 结束时间, 可选参数);

返回参数为:

```

data$Data 返回的序列数据, 为 data.frame 格式;
data$Code 无意义
data$ErrorCode 命令是否成功的错误码, 0表示成功。

```

实例: library(WindR)

```

w.start(showmenu=F); #不显示导航界面
w.tdays("2013-05-01", "2013-06-08") #返回 5 月 1 日到 6 月 8 日
之间的交易日序列

```

```
w.tdays("2013-05-01") #返回 5 月 1 日到当前时间的交易日序列
```

注:

- 1) 可选参数有很多种, 可选参数可以用数组实现;
- 2) 时间支持 R 中时间和日期格式, 结束时间缺省为当前时间;
- 3) 用户可以用 `w.menu('tdays')` 显示导航界面, 帮助创建命令;

2.18.2 w.tdaysoffset: 返回某个偏移值对应的日期

命令用来获取基于某个基准时间前推(<0)或者后推(>0)指定天数的日期。

用户可以使用 `?w.tdaysoffset` 查看命令说明。命令原型为: `data<-w.tdays(偏移值, 基准时间, 可选参数)`;

返回参数为:

```
data$Data 返回的序列数据, 为 data.frame 格式;  
data$Code 无意义  
data$ErrorCode 命令是否成功的错误码, 0表示成功。
```

实例: `library(WindR)`

```
w.start(showmenu=F); #不显示导航界面  
w.tdaysoffset(-5, "2013-05-01") #返回 5 月 1 日前推五个交易日的日期, 返回 2013-4-19  
w.tdaysoffset(-5) #返回当前时间前推五个交易日的日期
```

注:

- 1) 可选参数有很多种, 可选参数可以用数组实现;
- 2) 时间支持 R 中时间和日期格式, 基准时间缺省为当前时间;
- 3) 用户可以用 `w.menu('tdaysoffset')` 显示导航界面, 帮助创建命令;

2.18.3 w.tdayscount: 返回某个区间内日期数量

命令用来获取两个时间区间内的某种规则下的日期序列个数。用户可以使用 `?w.tdayscount` 查看命令说明。命令原型为: `data<- w.tdayscount(开始时间, 结束时间, 可选参数)`;

返回参数为:

```
data$Data 返回的序列数据, 为 data.frame 格式;  
data$Code 无意义  
data$ErrorCode 命令是否成功的错误码, 0表示成功。
```

实例：`library(WindR)`
`w.start(showmenu=F);` #不显示导航界面
`w.tdayscount("2013-05-01","2013-06-08")` #返回 5 月 1 日到 6 月 8 日之间的交易日序列长度，为 27
`w.tdayscount("2013-05-01")` #返回 5 月 1 日到当前时间的交易日序列长度

注：

- 1) 可选参数有很多种，可选参数可以用数组实现；
- 2) 时间支持 R 中时间和日期格式，结束时间缺省为当前时间；
- 3) 用户可以用 `w.menu('tdayscount')` 显示导航界面，帮助创建命令；

3 WinR 插件函数体说明

3.1 日期序列 (WSD)

函数名: `w.wsd (security, fields, startdate, enddate, option]`

返回选定证券品种的历史数据, 包括日间的行情数据, 基本面数据以及技术指标数据。

证券:			
Element	Element Value	Type	Description
Security		String	获取数据的证券列表
范例 1: '600030.SH' 说明: 证券列表支持 Wind 代码及证券转换类工具函数输出的 Wind 代码结果 只支持一个品种			
指标:			
Element	Element Value	Type	Description
Fields		String	获取数据的指标列表
范例 1: 'CLOSE,HIGH,LOW,OPEN' 范例 2: <code>c('CLOSE','HIGH','LOW','OPEN')</code>			
起始日期			
Element	Element Value	Type	Description
StartDate			时间序列的起始日期
范例 1: '2011-01-01','-5w', <code>Sys.Date()-5</code> 说明: 支持日期类工具函数输出的标准日期结果, 支持相对日期宏表达方式, 日期宏具体使用方式参考'日期宏'部分内容			
截止日期:			
Element	Element Value	Type	Description
EndDate			时间序列的截止日期, 若为空默认为系统当前日期
范例 1: '2011-06-30', <code>Sys.Date()</code> , 支持相对日期, 比如'0w'; 不输入的话为当前时间 说明: 支持日期类工具函数输出的标准日期结果, 支持相对日期宏表达方式			
指标参数 (可选):			
Element	Element Value	Type	Description
Parameter		String	提取指标时使用的参数名
Value		String	指定参数的值
范例 1: 'TRADE_DATE=20110301;FUND_DATE=20101231' 说明: 多指标参数支持在不同引号内分开取值			
变频参数 (可选):			

Element	Element Value	Type	Description
Period	D	String	每天一值
	W		每周一值
	M		每月一值
	Q		每季度一值
	S		每半年一值
	Y		每年一值
范例 1: 'Period=D' ，默认 Period=D			
输出日期（可选）：			
Element	Element Value	Type	Description
Days	Weekdays	String	所有工作日
	Alldays		所有日历日
	Trading		排除所有非交易日
范例 1: 'Days=Trading'，默认 Days=Trading			
填充方式（可选）：			
Element	Element Value	Type	Description
Fill	Previous	String	沿用之前数据
	Blank		返回空值
范例 1: 'Fill=Previous'，默认 Fill=Blank			
日期排序（可选）：			
Element	Element Value	Type	Description
Order	A	String	升序
	D		降序，最近日期在先
范例 1: ' Order =A'，默认 Order =A			
交易日历（可选）保留参数：			
Element	Element Value	Type	Description
TradingCalendar		String	选择不同交易所所在国家地区日历
范例 1: ' TradingCalendar =SSE'，默认 TradingCalendar =SSE;SSE 表示上交所，SZSE 表示深圳证券交易所，CFFE 表示中金所，DCE 表示大商所，CZCE 表示郑商所，SHFE 表示上期所，HKEX 表示香港交易所，TWSE 表示台湾证券交易所，Nasdaq 表示纳斯达克证券交易所，NYSE 表示纽约证券交易所，NYMEX 表示纽约商品交易所，COMEX 表示纽约金属交易所，NYBOT 表示纽约期货交易所，CME 表示芝加哥商业交易所。CBOT 表示芝加哥商品交易所，LME 表示伦敦金属交易所，IPE 表示伦敦国际石油交易所。			
输出币种（可选）：			
Element	Element Value	Type	Description
Currency	ORIGINAL	String	使用什么货币
	HKD		
	USD		
	CNY		

范例 1: 'Currency =Original', 默认 Currency =Original

3.2 历史截面数据 (WSS)

函数名: w.wss (security, fields, option)

返回选定品种的历史截面数据

证券：			
Element	Element Value	Type	Description
Security		String	获取数据的证券列表
范例 1： ' 600030.SH,600031.SH'			
指标：			
Element	Element Value	Type	Description
Fields		String	获取数据的指标列表
范例 1： ' LAST,HIGH,LOW,OPEN'			
指标参数（可选）：			
Element	Element Value	Type	Description
Parameter		String	提取指标时使用的参数名
Value		String	指定参数的值
范例 1： ' TRADE_DATE=20110301;FUND_DATE=20101231'			
输出币种（可选）：			
Element	Element Value	Type	Description
Currency	ORIGINAL	String	使用什么货币
	HKD		
	USD		
	CNY		
范例 1： ' Currency=CNY'，默认为 Currency=Original			

3.3 分钟序列 (WSI)

函数名: w.wsi (security, fields, starttime, endtime, option)

返回日内分钟 K 线数据, 包含当天;

证券:			
Element	Element Value	Type	Description
Security		String	获取数据的证券列表 说明: 只支持单市场单品种
范例 1: '600030.SH'			

指标：			
Element	Element Value	Type	Description
Fields		String	获取数据的指标列表
范例 1： ' CLOSE,HIGH,LOW,OPEN'			
起始日期时间			
Element	Element Value	Type	Description
StartTime			日内序列的起始日期和时间
范例 1： ' 2011-01-01 09:30:01'			
截止日期时间			
Element	Element Value	Type	Description
EndTime			日内序列的截止日期和时间 为空默认为最新时间
范例 1： ' 2011-01-01 14:30:01'			
指标参数（可选）：			
Element	Element Value	Type	Description
Parameter		String	提取指标时使用的参数名
Value		String	指定参数的值
范例 1： ' TRADE_DATE=20110301;FUND_DATE=20101231'			
间隔长短（可选）：			
Element	Element Value	Type	Description
BarSize	1-60	Long	在 1-60 间选择输入整数数字， 代表分钟数
范例 1： ' BarSize=1，默认 BarSize=1			
填充方式（可选）：			
Element	Element Value	Type	Description
Fill	Previous	String	沿用之前数据
	Blank		返回空值
范例 1： ' Fill=Previous'，默认 Fill=Blank			

3.4 日内跳价 (WST)

函数名: w.wst(security, fields, starttime, endtime, option)

返回日内盘口买卖十档快照数据和成交数据;

证券:			
Element	Element Value	Type	Description
Security		String	获取数据的证券列表

			说明：只支持单市场单品种
范例 1: '600030.SH '			
指标:			
Element	Element Value	Type	Description
Fields		String	获取数据的指标列表
范例 1: 'LAST_PRICE,HIGH,LOW,OPEN'			
起始日期时间:			
Element	Element Value	Type	Description
StartTime		String	跳价数据的起始日期和时间
范例 1: '2011-01-01 09:30:01'			
截止日期时间:			
Element	Element Value	Type	Description
EndTime		String	跳价数据的起始日期和时间
范例 1: '2011-01-01 09:50:01', 默认为系统当前最新时间			

3.5 实时数据 (WSQ)

函数名: w.wsq (security, fields, options = "", func = NULL)

返回选定品种的实时数据, 支持一次请求和订阅两种方式;

证券:			
Element	Element Value	Type	Description
Security		String	获取数据的证券列表
范例 1: '600030.SH,600031.SH'			
指标:			
Element	Element Value	Type	Description
Fields		String	获取数据的指标列表
范例 1: 'LAST,HIGH,LOW,OPEN'			
回调函数 (可选):			
Element	Element Value	Type	Description
func			指定回调函数
范例 1: w.demoCallback			

用户可以通过 demo (wsq_demo) 查看回调例子程序。

3.6 数据集 (WSET)

函数名：WSET，返回股票，基金，债券，商品等专题统计报表的数据。

数据集：			
Element	Element Value	Type	Description
VIEW		String	提取数据集的 VIEW 名
范例 1: 'SectorConstituent'			
View 参数（可选）：			
Element	Element Value	Type	Description
Parameter		String	提取指标时使用的参数名
Value		String	指定参数的值
范例 1: 'date=20130531';			
字段列表（可选）：			
Element	Element Value	Type	Description
FieldList		String	获取字段列表的数据
范例 1: 'sector=全部 A 股'			

3.7 条件选股 (WEQS)

函数名：WEQS，返回终端证券筛选的股票集。

数据集：			
Element	Element Value	Type	Description
filtername		String	终端条件选股的方案名
范例 1: '我的方案'，万得资讯终端上选股方案名为'我的方案'。			

3.8 资管函数 (WPF)

函数名：WPF，返回资产管理系统 AMS 统计报表的数据。

组合 ID/名称（必须）：			
Element	Element Value	Type	Description
Portfolio		String	提取数据集的组合 ID 或组合名称（在 AMS 系统中是产品名称）
范例 1: "武当一期"			
View 名称（必选）：			
Element	Element Value	Type	Description

VIEW		String	提取数据集的报表名称
范例 1: "PortfolioDaily" / "HoldingDaily"			
组合创建人 (可选):			
Element	Element Value	Type	Description
Owner		String	对 PMS 组合有效,当组合是别人共享的,在此给出该组合的创建人 Wind 帐号
范例 1: "OWNER=W08000001"			
View 参数 (可选):			
Element	Element Value	Type	Description
Parameter		String	提取报表时使用的参数名
Value		String	指定参数的值
范例 1: "TRADE_DATE=20110301;CURRENCY=CNY"			
字段列表 (可选):			
Element	Element Value	Type	Description
FieldList		String	获取字段列表的数据
范例 1: "FIELD=port_name,port_id"			

3.9 组合上传函数 (wupf)

函数名:WUPF, 投资者将自己的每天持仓数据上传至终端 PMS 中。

wupf(PortfolioName, TradeDate, WindCodes, Quantity, CostPrice, [optional arguments])

组合 ID/名称 (必选, 输入) :			
Element	Element Value	Type	Description
PortfolioName		Variant	欲上传持仓的组合名称, 必须为已有组合
范例 1: "武当一期"			
持仓日期 (必选, 输入) :			
Element	Element Value	Type	Description
TradeDate		Variant	持仓日期, 当上传多个持仓的时候可以输入数组
范例 1: "20131231"			
持仓品种 (必选, 输入) :			
Element	Element Value	Type	Description
WindCodes		Variant	持仓品种代码, 当上传多个持仓的时候可以输入数组。 现金视为证券的一种, 现金数量为其金额, 价格为 1, 目前仅支持单一现金, 不支持多现金账户。 其中现金代码如下: CNY 为人民币 USD 为美元 HKD 为港币
范例 1: "600000.SH"			
持仓量 (必选, 输入) :			
Element	Element Value	Type	Description
Quantity		Variant	持仓数量, 股票为股, 期货为手, 现金为其数额, 必须为整数
范例 1: "200"			

持仓成本（必选，输入）：			
Element	Element Value	Type	Description
CostPrice		Variant	持仓成本价格（含佣金等交易费用），可以为负数，默认为当日收盘价。现金价格为 1。
范例 1: "15.8"			
组合创建人（可选，输入）：			
Element	Element Value	Type	Description
Owner		String	组合拥有者的 Wind 用户账号，默认为当前账户
范例 1: "OWNER=W0800001"			
说明：当组合是别人共享的，填入组合创建人的 Wind 帐号			
买卖方向（可选，输入）：			
Element	Element Value	Type	Description
Direction	Long	String	多方
	Short	String	空方
范例 1: "Direction=Long "			
说明：默认为 Long			
证券类型（可选，输入）：			
Element	Element Value	Type	Description
AssetType	Margin	String	融资融券
	Cash	String	现金
	Equity	String	股票
	Bond	String	债券
	Repo	String	债券回购
	Fund	String	基金
	Cmdty	String	期货
	SFP	String	券商理财产品
	Trust	String	信托产品
	BFP	String	银行理财产品
	Pfund	String	阳光私募
范例 1: " AssetType=Margin"			

说明：后台可以自动解析资产类别，因此除融资融券字段外，无需设置相应类别。一旦在此设置，后台不做类别错误检查。

投机套保类型（可选，输入）：

Element	Element Value	Type	Description
HedgeType	Spec	String	投机
	Hedge	String	套保
范例 1: "HedgeType=Spec"			
说明：是否为投机与套保，默认为投机。如果选择套保需要专门的套保账号。			

3.10 交易函数

3.10.1 登录 (tlogin)

函数名：tlogin 可以登录资金账号或者模拟账号。登陆成功时系统自动生成一个登录号。

经纪商代码（必须）：			
Element	Element Value	Type	Description
BrokerID		String	经纪商的代码，每家经纪商都有一个编码。
范例 1: " '0000' " 即 WTTS 模拟柜台			
营业部代码（必选）：			
Element	Element Value	Type	Description
Department ID		String	券商营业部代码
范例 1: "0", 0 表示不必填写。			
资金账号（必选）：			
Element	Element Value	Type	Description
AccountID		String	资金账号
范例 1: " 000100000090" #WFT 用户模拟账号期货为账号+02，股票为账号+01			
资金密码（必选）：			
Element	Element Value	Type	Description
Parameter		String	提取报表时使用的参数名
Password		String	资金账号密码

范例 1: "aaa", #WFT 用户模拟账号密码初始值为 000000

字段列表（必选）：

Element	Element Value	Type	Description
AccountType		String	账户类型，其含义如下。 SH、SZ、SHSZ 深圳上海 A SZB 深圳 B SHB 上海 B CZC 郑州商品 SHF 上海商品 DCE 大连商品 CFE 股指商品

范例 1: "SZSH"

3.10.2 登出 (tlogout)

函数名：tlogout，退出登录号。

登录 ID（单账号登录可选，多账号登录时）：

Element	Element Value	Type	Description
Logonid		String	登录号。

范例 1: " '0000' "

3.10.3 下单 (torder)

函数名：torder，委托下单。

Wind 码（必选）：

Element	Element Value	Type	Description
SecurityCode		String	Wind 代码。也可以直接输入交易代码，但此时需要提供 MarketType

范例 1: " 600000.SH "

交易方向（必选）：

Element	Element Value	Type	Description
TradeSide		String	交易方向 Buy '1' //买入开仓 (等同=证券买入) Short '2' //卖出开仓 Cover '3' //买入平仓 Sell '4' //卖出平仓 (等同=证券卖出) CoverToday '5' //买入平今仓

			SellToday '6' //卖出平今仓
范例 1: "Buy"或者"1"			
委托价格 (必选) :			
Element	Element Value	Type	Description
OrderPrice		Double	买卖价格
范例 1: 6.8			
交易数量 (必选) :			
Element	Element Value	Type	Description
OrderVolume		Int	买卖数量
范例 1: 100			
价格委托方式 (可选) :			
Element	Element Value	Type	Description
OderType		String	委托方式, 默认为限价交易。 LMT 0 //限价委托 BOC 1 //best of counterparty.对方最优价格委托 BOP 2 //best of party.本方最优价格委托 ITC 3 //immediately then cancel.即时成交剩余撤销 B5TC 4 //best 5 then cancel.最优五档剩余撤销 FOK 5 //fill or kill.全额成交或撤销委托 B5TL 6 //best 5 then limit.最优五档剩余转限价 //目前, 深圳支持的方式为0-5, 上海只支持0, 4、6三种
范例 1: OderType ="LMT" 表示限价委托			
套保标志 (可选) :			
Element	Element Value	Type	Description
HedgeType		Int	是否为投机套保。确实为 SPEC 投机, 如果选择套保需要专门的套保账号。 SPEC '0' //'0'-投机 HEDG '1' //'1'-保值
范例 1: HedgeType=SPEC			
登录 ID (单账号登录时可选, 多账号时必选) :			
Element	Element Value	Type	Description
LogonID		Int	登录号。
范例 1: logonID=4			
市场类型 (可选) :			
Element	Element Value	Type	Description

MarketType		String	当输入的是交易代码不是 Wind 码时，需要输入市场代码。		
			SZ	0	//证券-深圳
			SH	1	//证券-上海
			OC	2	//证券-深圳特（三版）
			HK	6	//证券-港股
			CZC	7	//商品期货 (郑州)
			SHF	8	//商品期货 (上海)
			DCE	9	//商品期货 (大连)
			CFE	10	//股指期货 (中金)
范例 1: MarketType='SH'					

3.10.4 撤单 (tcancel)

函数名: tcancel 取消委托。

委托号（必选）：			
Element	Element Value	Type	Description
OrderNumber		String	委托号。
范例 1：22 表示委托号是 22。委托号可以通过 w.tquery(2) 得到。			
市场类型（可选）：			
Element	Element Value	Type	Description
MarketType		String	市场类型。当 OrderNumber 存在重复时必须填。
			SZ0//证券-深圳
			SH1//证券-上海
			OC2//证券-深圳特（三版）
			HK6//证券-港股
			CZC7//商品期货（郑州）
			SHF8//商品期货（上海）
			DCE9//商品期货（大连）
			CFE10//股指期货（中金）
范例 1：MarketType = `SH`			
登录 ID（单账号登录时可选）：			
Element	Element Value	Type	Description
Logonid		String	登录号
范例 1：LogonID=3			

3.10.5 查询 (tquery)

函数名: tquery 查询。

查询内容（必选）：			
Element	Element	Type	Description

	Value		
qrycode		String	查询字段含义如下: 0 Capital 资金查询 1 Position 持仓查询 2 Order 当日委托查询 3 Trade 当日成交查询 4 Department 营业部查询 5 Account 查询股东账号, 或者期交所查询 6 Broker 经纪商查询 7 LogonID 登录号查询
范例 1: "order"			
登录号 LogonID (可选; 多账号时, qrycode=0-3, 5 时必选):			
Element	Element Value	Type	Description
LogonID		String	登录号。
范例 1: LogonID=0			
请求号查询 (可选):			
Element	Element Value	Type	Description
RequestID		String	系统生成请求号。Qrycode='Order' / 2 有意义。立即返回本地委托状态
范例 1: RequestID='12'			
委托号 (可选):			
Element	Element Value	Type	Description
OrderNumber		String	委托号。Qrycode='Order' / 'Trade' 有意义。
范例 1: OrderNumber='12'			
股票代码查询 (可选):			
Element	Element Value	Type	Description
WindCode		String	Wind 代码; Qrycode='Position'、'order'、'Trade' 有意义。
范例 1: 'WindCode=002311.SZ'			
经纪商 ID (可选):			
Element	Element Value	Type	Description
BrokerID		String	Qrycode='Department' 有意义。
范例 1: BrokerID='0000'			

3.11 日期函数

3.11.1 特定交易日 (TDAYS)

函数名: TDays(startDate,endDate,[Optional argument])

释义: TradingCalendar 指定特定交易所交易日, 从 StartDate 到 EndDate

交易日 (或日历日) 的列表, Period 按照周期返回日期序列

起始日期			
Element	Element Value	Type	Description
StartDate		String	时间序列的起始日期
范例 1: "2011-01-01", 支持日期宏			
截止日期:			
Element	Element Value	Type	Description
EndDate		String	时间序列的截止日期, 置空取当前最新日期
范例 1: "2011-06-30", 支持日期宏			
日期类型 (可选)			
Element	Element Value	Type	Description
Days	Weekdays	String	工作日
	Alldays		日历日
	Trading		交易日
范例 1: "Days=Trading", 默认"Days=Trading"			
变频参数 (可选) :			
Element	Element Value	Type	Description
Period	D	String	每天一值
	W		每周一值
	M		每月一值
	Q		每季度一值
	S		每半年一值
	Y		每年一值
范例 1: "Period=D" , 默认 Period=D			
交易日历 (可选)			
TradingCalendar 默认为上海证券交易所, 当 DAYS 为日历日的时候, 这个参数不起作用, 只有当 DAYS 为交易日的时候, 这个参数才起作用 默认"TradingCalendar=SSE"(上海证券交易所)			

3.11.2 日期偏移函数 (TDAYSOFFSET)

函数名 TDaysOffset(offset, refDate, [Optional argument])

释义：TradingCalendar 指定特定交易所交易日，从 refDate 起，Offset (偏移，>0 后推，<0 前推) 个 Period (周期) 的日期

参考日期			
Element	Element Value	Type	Description
refDate		String	参照日期
范例 1: "2011-01-01"			
日期类型 (可选)			
Element	Element Value	Type	Description
Days	Weekdays	String	工作日
	Alldays		日历日
	Trading		交易日
范例 1: "Days=Trading", 默认 "Days=Trading"			
变频参数 (可选) :			
Element	Element Value	Type	Description
Period	D	String	每天一值
	W		每周一值
	M		每月一值
	Q		每季度一值
	S		每半年一值
	Y		每年一值
范例 1: "Period=D" , 默认 Period=D			
TradingCalendar (可选)			
TRADINGCALENDAR 默认为上海证券交易所, 当 DAYS 为日历日的时候, 这个参数不起作用, 只有当 DAYS 为交易日的时候, 这个参数才起作用 默认 TradingCalendar=SSE			
偏移量 (可选)			
Element	Element Value	Type	Description
Offset			
偏移参数, 为整数, >0 后推, <0 前推, 默认为 0			

3.11.3 交易日统计 (TDAYSCOUNT)

函数名: TDaysCount(startDate, endDate, [Optional argument])

释义:TradingCalendar 指定特定交易所交易日,从 StartDate 到 EndDate

交易日 (或日历日) 总数

起始日期			
Element	Element Value	Type	Description
StartDate		String	起始日期
范例 1: "2011-01-01"			
截止日期:			
Element	Element Value	Type	Description
EndDate		String	截止日期
范例 1: "2011-06-30"			
日期类型 (可选)			
Element	Element Value	Type	Description
Days	Weekdays	String	工作日
	Alldays		日历日
	Trading		交易日
范例 1: "Days=Trading", 默认"Days=Trading"			
交易日历 (可选)			
TradingCalendar 默认为上海证券交易所, 当 DAYS 为日历日的时候, 这个参数不起作用, 只有当 DAYS 为交易日的时候, 这个参数才起作用			
默认 TradingCalendar=SSE			

3.12 日期宏

3.12.1 通用日期宏

支持相对日期表达方式, 相对日期周期包括: TD/D/W/M/Q/S/Y, 即交易日/日

历日/日历周/日历月/日历季/日历半年/日历年

➤ 以'-'代表前推, 数字代表 N 个周期, 只支持整数; 后推没有负号;

比如'-5D'表示从当前最新日期前推 5 个日历日;

➤ 截止日期若为''空值, 取系统当前日期;

- 可对日期宏进行加减运算，比如‘ED-10d’；

举例：

1. 起始日期为 1 个月前，截至日期为最新

StartDate=' -1W',EndDate=' '

2. 起始日期为前推 10 个交易日，截至日期为前推 5 个交易日

StartDate=' -10TD',EndDate=' -5TD'

3.12.2 特殊日期宏

目前条件选股，数据浏览器中有许多日期宏，数据接口支持如下日期宏：

宏名称	宏助记符
截止日期	ED
去年一季	LQ1
去年二季	LQ2
去年三季	LQ3
去年年报	LYR
今年一季	RQ1
今年二季	RQ2
今年三季	RQ3
最新一期	MRQ
本年初	RYF
下半年初	RHYF
本月初	RMF
本周一	RWF
上周末	LWE
上月末	LME
上半年末	LHYE
上年末	LYE
上市首日	IPO

4 WindR 应用案例

4.1 提取数据

4.1.1 提取历史交易报价

例：提取银行间交易债券 09 付息国债（090007.IB）的净价序列数据，时间从 2012-1-1 到最新。

```
#启动WindR接口;
library(WindR)
w.start()
#设置起始时间和截止时间，通过wsd接口提取序列数据
begintime<-'20120101';
endtime<-Sys.Date();

wdata<-w.wsd('090007.IB','close',begintime,endtime,'Priceadj=CP;tradingcalendar=NIB');
```

其中，'Priceadj', 'CP'表示债券净价，'U'表示不对股票除权，
'tradingcalendar', 'NIB'为银行间市场交易日历。

例：提取 000001.SZ 的开高低收数据，起始时间前推 100 天（日期宏），截止时间最新，前复权数据。

```
#启动WindR接口;
library(WindR)
w.start()
#设置起始时间和截止时间，通过wsd接口提取序列数据
begintime<-'20120101';
endtime<- Sys.Date();

wdata<-w.wsd('000001.SZ','open,high,low,close','-100d',endtime,'Priceadj=F');
```

其中，-100d 是日期宏函数，表示前推 100 天。

4.1.2 提取分钟序列数据

例：提取中金所 IF 股指期货当月连续合约的 3 分钟数据，截止时间最新（Sys.time()），起始时间前推 100 天（Sys.Date()-100）；

```
#启动WindR接口;
library(WindR)
w.start()
#设置起始时间和截止时间，通过wsi接口提取序列数据
codes='IF00.CFE';
fields='open,high,low,close';
begintime= Sys.Date()-100;
endtime= Sys.time()
```

```
wdata= w.wsi(codes,fields,begintime,endtime,'BarSize=3');
```

其中Sys.time()是R内置的日期函数，表示当前时刻。

4.1.3 提取盘口买卖盘数据

例：提取平安银行（000001.SZ）当天的买卖盘数据：

```
#启动WindR接口;
library(WindR)
w.start()
#设置起始时间和截止时间，通过wsi接口提取序列数据
begintime=format(Sys.time(),'%Y%m%d 09:30:00');
endtime =Sys.time();
codes='000001.SZ'
#last最新价, amt成交额, volume成交量
#bid1 买1价, bsize1 买1量
#ask1 卖1价, asize1 卖1量
fields='last,bid1,ask1';
wdata <- w.wst(codes,fields,begintime,endtime);
```

4.1.4 提取截面数据

例：提取浦发银行（600000.SH）、万科 A（000002.SZ）、宝安 A（000009.SZ）、南玻 A（000012.SZ）、长城开发（000021.SZ）2012 年 11 月 30 号的基本特征字段，包括公司名称、公司英文名称、IPO 日期、流通股、净流入资金、流入量，相应的字段为 comp_name, comp_name_eng, ipo_date, float_a_shares, mf_amt, mf_vol。

```
#启动WindR接口;
library(WindR)
w.start()
codes='600000.SH,000002.SZ,000009.SZ,000012.SZ,000021.SZ';
fields='comp_name,comp_name_eng,ipo_date,float_a_shares,mf_amt,mf_vol';
wdata<- w.wss(codes,fields,'tradedate=20121130');
```

其中，'tradedate'表示交易日期。

4.1.5 提取实时行情数据

例：提取世纪星源（000005.SZ）、深振业（000006.SZ）、零七股份（000007.SZ）、宝利来（000008.SZ）股票的当日实时指标数据：

```
library(WindR)
w.start()
w_data=w.wsq('000005.SZ,000006.SZ,000007.SZ,000008.SZ','rt_time,rt_last,rt_bid1,rt_ask1,rt_vwap');
```

其中，rt_time,rt_last,rt_bid1,rt_ask1,rt_vwap 分别为时间、现价、买入价、

卖出价、成交均价字段。

4.1.6 提取财务数据

例：提取海正药业（600276.SH）、恒瑞医药（600276.SH）、双鹭药业（002038.SZ）、天士力（600535.SH）2012 年年报中的营业收入、营业利润、净利润数据，数据来源为合并报表。

```
library(WindR)
w.start()
wdata=w.wss('600276.SH,600276.SH,002038.SZ,600535.SH','oper_rev,opprofit,net_profit_is','rptDate=20121231;rptType=1')
```

其中，营业收入、营业利润、净利润对应的字段为 oper_rev、opprofit、net_profit_is，报告期为 2012 年 12 月 31 日（rptDate=20121231），财务报表为合并报表（rptType=1）。

4.1.7 提取债券估值数据

例：提取银行间国债 09 年 07 附息券（090007.IB）的全价、应计利息、估价修正久期，数据来源为中证指数公司，对应的字段为 dirty_csi、accruedinterest_csi、modidura_csi。日期为 2013 年 4 月 6 日至 5 月 6 日。

```
library(WindR)
w.start()
wdata=w.wsd('090007.IB','dirty_csi,accruedinterest_csi,modidura_csi','2013-04-06','2013-05-06')
```

注意，目前支持中债公司、中证指数公司、清算所的债券估价，中债公司需要取得授权，清算所的债券估值数据较少。

4.1.8 提取数据集

目前可以读取板块成分、指数成分股及权重、ETF 申赎成分信息、分级基金明细、融资标的、融券标的、融资融券担保品、回购担保品、停牌股票、分红送转等股票数据。

例：先读取 HS300 成分股指数的权重，日期为 2013 年 6 月 3 日。然后提取融资融券标的余额与资金流入流出数据。

```
library(WindR)
w.start()
```

```
data<-w.wset('IndexConstituent','date=20130603;windcode=000300.SH')
```

```
#下面读取沪深 2 市融资融券标的余额与资金流入流出统计。
```

```
# 变量说明：
```

```
# %% 融资标的代码
```

```
# w_wset_data1 融资标的代码
```

```
# w_wset_data2 融券标的代码
```

```
# %% 融资余额统计
```

```
# MarginBuy1;% 融资买入额
# MarginBuy2;% 融资偿还额
# MarginBuy3;% 融资余额
# %% 融券余额统计
# MarginSell1;% 融券卖出量
# MarginSell2;% 融券偿还量
# MarginSell3;% 融券余量
# MarginSell4;% 融券余额
# %% 融资品种资金流入统计
# buyCash1;% 净流入资金
# buyCash2;% 净流入量
# buyCash3;% 金额流入率
# buyCash4;% 资金流向占比
# buyCash5;% 尾盘净流入资金
# buyCash6;% 开盘净流入资金
# %% 融券品种资金流入统计
# SellCash1;% 净流入资金
# SellCash2;% 净流入量
# SellCash3;% 金额流入率
# SellCash4;% 资金流向占比
# SellCash5;% 尾盘净流入资金
# SellCash6;% 开盘净流入资金

#设定开始时间与结束时间
BeginDay = '2012-06-01'
EndDay   = '2013-05-28'

# 读取融资标的
# 读取融资标的
w_wset_data1<-w.wset('MarginTradingUnderlying','date=20130530');
#% 1.2 融资标的余额
i=2
data<-w.wsd(w_wset_data1$Data[[3]][i],'mrg_long_amt,mrg_long_repay,mrg_long_bal',BeginDay,EndDay);

# 融资标的资金流向
data<-w.wsd(w_wset_data1$Data[[3]][i],'mf_amt,mf_vol,mf_amt_ratio,mf_vol_ratio,mf_amt_close,mf_amt_open',BeginDay,EndDay);

# 读取融券标的
w_wset_data2<-w.wset('ShortSellingUnderlying','date=20130530');
# 融券标的余额统计
data<-w.wsd(w_wset_data1$Data[[3]][i],'mrg_short_vol,mrg_short_vol_repay,mrg_short_vol_bal,mrg_short_bal',BeginDay,EndDay);
```

```
# 融券余额资金流向
data<-w.wsd(w_wset_data1$Data[[3]][i], 'mf_amt, mf_vol, mf_amt_ratio
, mf_vol_ratio, mf_amt_close, mf_amt_open', BeginDay, EndDay);
```

4.1.9 提取资管报表数据

例：某用户终端中资管中选择了名为“130325”的组合，现在将该组合的统计数据读出来，选择的报表为“组合结算数据”，报表字段为：Portfolio_Name（组合名称）、Portfolio_ID（组合ID）、Total_Asset（总资产）

```
library(WindR)
w.start()

data=w.wpf('130325', 'PMS.PortfolioDaily', 'startdate=20130503;enddate=20130603;reportcurrency=CN¥;owner=;field=Portfolio_Name, Portfolio_ID, Total_Asset')
```

4.1.10 提取交易日期

例：提取上海期货交易所 2013 年 5 月 3 日至 6 月 3 日的交易日期

```
library(WindR)
w.start()
data=w.tdays('2013-05-03', '2013-06-03', 'TradingCalendar=SHFE;')
%其中, 'TradingCalendar=SHFE;' 是上海期货交易所代码，默认是上海证券交易所。
```

例：提取上海股票交易所 2013 年 6 月 3 日前推 4 个交易日的日期。

```
library(WindR)
w.start()

data=w.tdaysoffset(-4, '2013-06-03')
```

例：统计上海证券交易所交易日期

```
library(WindR)
w.start()

data=w.tdayscount('2013-05-03', '2013-06-03')
```

4.2 读取股票日 K 线价格并绘制价格图

【例 8】读取恒瑞医药（600276.SH）历史收盘价，时间是从 2013 年 1 月 2 日至 2013 年 4 月 2 日，并绘制各种股票价格图。

StockList='600276.SH' ; #注意目前历史数据每次仅能读取一只股票。

读取股票收盘价（2013 年 4 月 2 日）

```
library(WindR)
require(quantmod)
```

```
w.start()
wdata=w.wsd(StockList,'open,high,low,close,volume','2013-01-02','
2013-04-02');
data <-wdata$Data
ts <- xts(data[,-1],data[,1])
chartSeries(ts,TA=c(addVo(),addBBands(),addMACD()),up.col="red",d
n.col="#00ffff",name=StockList)
```



4.3 Demo 程序介绍

用户可以使用 `demo(package='WindR')` 列出 WindR 附带的 demo 程序

也可以使用 `?WindR`, 然后通过帮助界面的底部 `index` 链接得到 demo 帮助,

参见 1.4.2

4.3.1 wsd_quant_demo

该实例需要 `quantmod` 安装包, 文件内容如下, 用户可以使用 `demo(wsd_quant_demo)` 运行, 运行过程中注意敲回车。

```
require(WindR)
require(quantmod)

#user should start WindR firstly.
w.start(showmenu=FALSE);
```



```

code<-"600004.SH"
wsd_data<-
w.wsd(code,"open,high,low,close,volume",Sys.time()-24*3600*100,Sys.time()-24*3600)
if(wsd_data$ErrorCode[[1]]!=0)
{
  error("w.wsd error")
}

data <-wsd_data$Data
ts <- xts(data[,-1],data[,1])
chartSeries(ts,TA=c(addVo(),addBBands(),addMACD()),up.col="red",dn.col="#00ffff",name=code)

```

输出界面见 4.2

4.3.2 wsi_demo

文件内容如下，用户可以使用 `demo(wsi_demo)` 运行，运行过程中注意敲回车。该实例优点是绘图部分是自己的，没有用别的包，便于控制。

```

require(WindR)

w.start(showmenu=FALSE);

code<-"600004.SH"
wsi_data<-
w.wsi(code,"open,high,low,close",Sys.time()-24*3600*8,Sys.time(),options='BarSize=10')
if(wsi_data$ErrorCode[[1]]!=0)
{
  error("w.wsd error")
}

data<-wsi_data$Data;

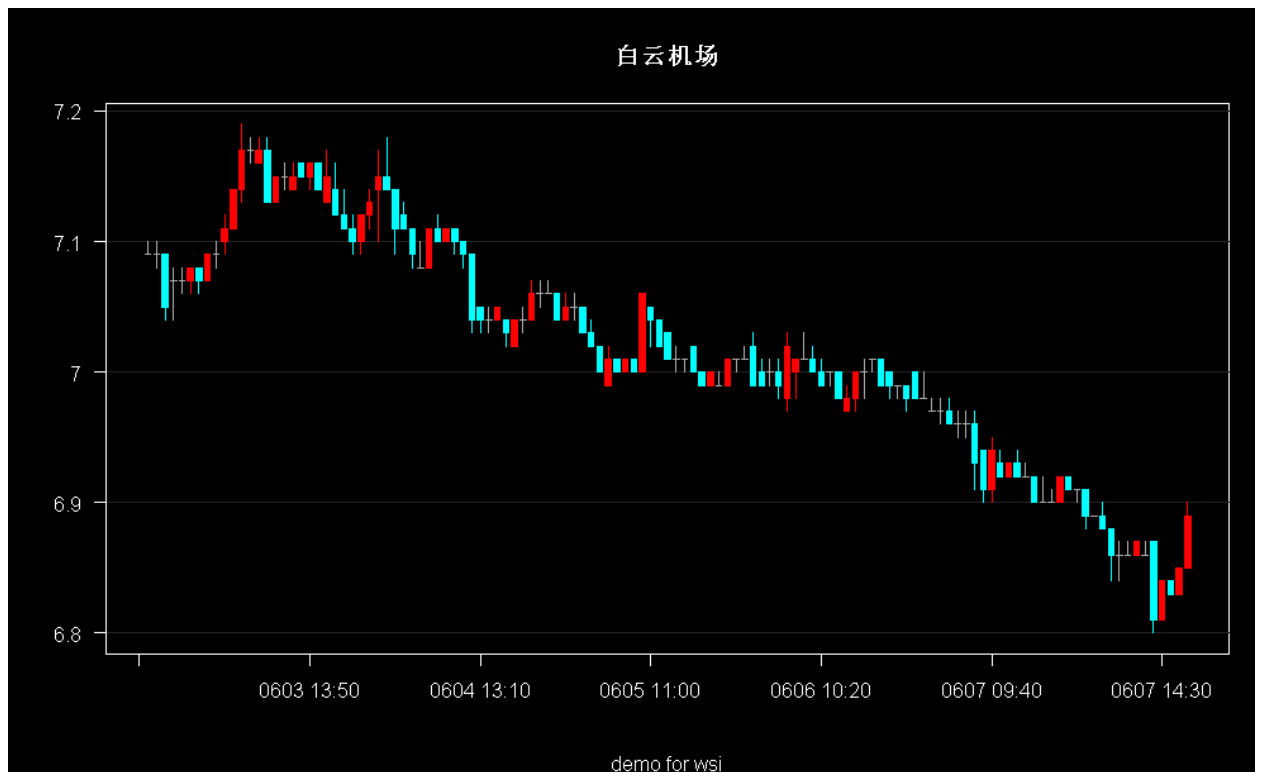
getnamebycode<-function(code)
{
  data<-w.wss(code,"sec_name") #通过 WSS 获取股票名称
  if(data$ErrorCode==0)
    return (as.character(data$Data$SEC_NAME[[1]]))
  else
    return (NULL)
}

```

```
}

#...
#具体绘图功能请直接查看源文件
```

输出界面见为下图:



4.3.3wst_demo

文件内容如下, 用户可以使用 `demo(wst_demo)` 运行, 运行过程中注意敲回车。该实例优点是绘图部分是自己的, 没有用别的包, 便于控制。

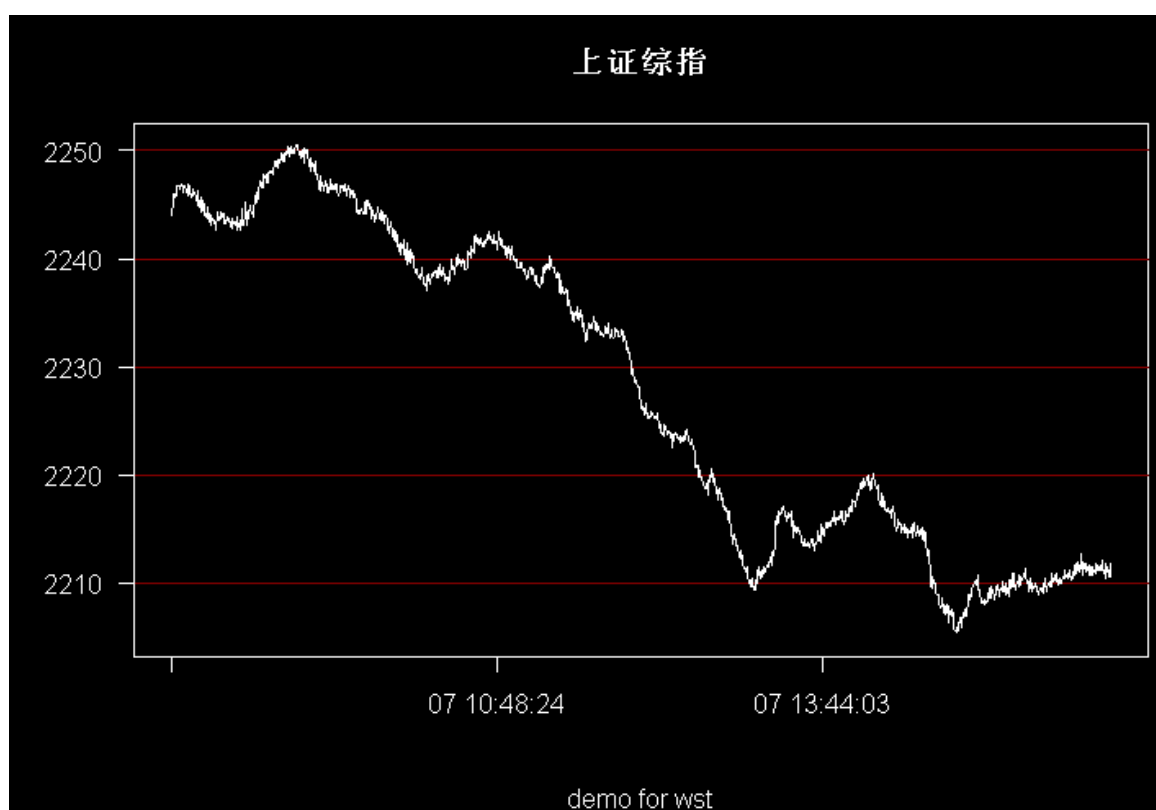
```
require(WindR)
w.start(showmenu=FALSE);

code<-"000001.SH";#IF1306.CFE";#600004.SH"
begintime<-format(Sys.time(),"%Y%m%d 09:30:00");
endtime <-format(Sys.time(),"%Y%m%d 15:00:00");

wst_data<- w.wst(code,"open,high,low,last",begintime,endtime)
if(wst_data$ErrorCode[[1]]!=0)
{
  error("w.wsd error")
}
data<-wst_data$Data;
```

```
getnamebycode<-function(code)
{
  data<-w.wss(code,"sec_name")#通过 wss 获取股票名称
  if(data$ErrorCode==0)
    return (as.character(data$Data$SEC_NAME[[1]]))
  else
    return (NULL)
}
#...
#具体绘图功能请直接查看源文件
```

输出界面见为下图：



4.3.4wsq_demo

文件内容如下，用户可以使用 `demo(wsq_demo)` 运行，运行过程中注意敲回车。该实例用 R 实现了实时 5 档行情报价显示界面。用户需要停止实时界面时应使用 `stopwsq()` 命令

```
require(WindR)
require(graphics)
```

```
#...
#具体绘图、初始化功能请直接查看源文件

getnamebycode<-function(code)
{
  data<-w.wss(code,"sec_name")
  if(data$ErrorCode==0)
    return (as.character(data$Data$SEC_NAME[[1]]))
  else
    return (NULL)
}

wsqcallback<-function(data)
{#回调函数主体
  # data 包含如下信息
  # $RequestID 订阅请求 ID
  # $Field 数据中对应的指标名
  # $Code 数据中对应的代码
  # $Time 返回数据对应的时间.
  # $ErrorCode 函数运行的错误 ID.
  # $Data 返回的数据结果, 为三维数组, 对应指标 Field、代码 Code、时间
Time 三个维度
  if(data$ErrorCode!=0)
  {
    return()
  }

  if(length(data$Code)!=1 || data$Code[[1]]!=gStockData$CODE)
  {
    return ();
  }

  updatename<-unlist(data$Field);
  gStockData[updatename]<<-unlist(data$Data)

  #....
  #.....draw.....
  drawbg()
  drawvalues()
  print("update...")
}

startwsq<-function()
```

```
{#启动应用
```

```
#使用 WSQ 订阅
```

```
data<-w.wsq(gStockData$CODE,"rt_high,rt_low,rt_vwap,rt_open,rt_a  
mt,rt_swing,rt_upward_vol,rt_downward_vol,rt_last_vol,rt_vol_rat  
io,rt_vol,rt_turn,rt_bsize5,rt_bsize4,rt_bsize3,rt_bsize2,rt_bsi  
zel,rt_bid5,rt_bid4,rt_bid3,rt_bid2,rt_bid1,rt_ask5,rt_ask4,rt_a  
sk3,rt_ask2,rt_ask1,rt_asize5,rt_asize4,rt_asize3,rt_asize2,rt_a  
size1,rt_last,rt_pre_close,rt_chg,rt_pct_chg" ,func=wsqcallback)  
  if(data$ErrorCode!=0)  
  {  
    print("call wsq error!")  
    return()  
  }  
  
  gStockData$requestID<-data$requestID;  
  print(gStockData$requestID)  
}
```

```
stopwsq<-function()
```

```
{#启动应用
```

```
  if(gStockData$requestID[[1]]!=0)  
    w.cancelRequest(gStockData$requestID)  
  gStockData$requestID<-0;  
}
```

```
#.....请直接看源代码
```

界面如下图，在交易时会动态变化。

浦发银行		600000
9.3500		-0.1000 -1.06%
Ask 5	9.40	3003
Ask 4	9.39	4942
Ask 3	9.38	1637
Ask 2	9.37	1201
Ask 1	9.36	1421
Bid 1	9.35	136
Bid 2	9.34	592
Bid 3	9.33	5143
Bid 4	9.32	5696
Bid 5	9.31	7409
Total Lots	723441	Turnover 0.48%
Current Lots	218	Vol. Ratio 0.967
OutSize	397803	InSize 325638
Total	68201	Amplitude 2.65%
Avg Price	9.43	Open 9.51
HIGH	9.57	LOW 9.32

5 常见问题

5.1 安装及注册

Q: 点击量化菜单中“量化”选项，提示“找不到 R”

检查一下本地电脑是否已经安装了 R 软件（版本大于 R2.15.0）。R 软件

是免费软件，用户可以到<http://www.r-project.org/>下载该软件。

有的 R 环境没有正常安装，此时需要使用 1.2.4 介绍方法安装。

Q: WindR 插件支持的 R 版本？ 64 位是否支持？

WindR 支持 2.15.0 以上版本。

WindR 插件支持 64 位。

Q: WindR 注册出现错误原因如下。

A: 检查 R 环境是否退出？ 如果 Matlab 环境正在使用 Wind 插件也请关闭 Matlab。

B: 检查 R 版本是否为 R2.15.0 以后。检查 R 环境是否是免安装版本。

C: 检查 Wind 资讯终端是否升级。注意 Wind 资讯终端不能安装在中文目录下。

D: 资讯所在公司的 IT 管理员，申请取得“管理员权限”

Q: 报 failed to lock directory. ... for modifying. ... try removing ... 00LOCK-WindR

请删除 00LOCK-WindR 再试。

5.2 读取指标数据

Q: WindR 读取数据步骤：

WindR 读取数据前一定要运行下面代码。

```
>> w.start()
```

菜单向导如下。

```
>>w.menu()
```

WindR 读取数据通过下面 7 个函数实现的。

`w.wsd` 读取历史序列数据,包括日间的行情数据,基本面数据以及技术数据指标。

`w.wss` 读取股票、债券、商品等的基本面静态数据。

`w.wst` 盘口买卖十档快照数据和分时成交数据。

`w.wsi` 读取分钟级别历史及当天行情数据。

`w.wsq` 读取证券实时行情数据及技术指标。

`w.wset` 读取板块成份、指数成份权重等数据。

`w.wpf` 获取资产管理、组合管理相关信息。

`w.tdays` 返回区间内的日期序列

`w.tdaysoffset` 返回某个偏移值对应的日期

`w.tdayscount` 返回某个区间内日期数量

具体参考 Wind 自带的例子。

Q: WindR 能取哪些指标呢?

建议在 R 中运行下面代码。

```
>> w.start()
```

或者>>w.menu()

随后会弹出对话框,对话框中的指标都是可以读取的。

Q: WindR 能取哪些指标呢? 能否取财务等基本面指标?

可以读取股票及债券基本面数据、基本行情数据,财务数据。更多的数据正在整理中,预计很快可以对用户放开。

Q: WindR 是否可以判断股票是否是 ST 股?

通过风险 `w.wset` 命令实现。具体来说通过 `w.menu('wset')` 调出 `wset` 导航界面，然后选择“板块成分”，选择“板块名称”，选择“沪深股票”，“风险警示股票”。

对应的语句为：
`data<-w.wset("SectorConstituent","date=20130609;sector= 风险警示股票;field=wind_code")`

然后可以用 `any(data$Data$wind_code=='900955.SH')` 语句判断。

Q: WindR 是否可以提取多品种多指标?

序列数据接口仅支持单品种多指标，包括 `WSD`、`WSI` 和 `WST`；界面数据接口支持多品种多指标，包括 `WSS` 和 `WSQ`。

Q: 数据提取超限后怎么处理?

目前 `wsd`、`wsi`、`wst` 只能取单品种数据，不支持多品种。如出现数据提取超限后（`$ErrorCode=-40522017`）请电话万得客服电话（400-820-9463），请求增加提取数据的权限，并说明使用的函数名。

Q: 如何使用日期宏?

日期宏函数主要是便于日期提取，例如日期宏“TD”是 `TradeDate` 缩写，表示交易日，`-100TD` 表示前 100 个交易日，`100TD` 表示后 100 个交易日。提取 2013 年 4 月 8 日前 100 个交易日收盘价数据可以使用如下命令实现。

```
w_wsd_data=w.wsd('000001.SZ','close','-100TD','2013-04-08')
```

其中“-100TD”代表以当前时间为基准的前 100 个交易日。

```
w_wsd_data=w.wsd('000001.SZ','close','ED-100TD','2013-04-08')
```

其中 ED 表示以结束时间为基准，'ED-100TD'代表以结束时间为基准的前 100 个交易日

Q: 为什么分钟线没有 11:30? 为什么 5 分钟线没有 9:30 ?

分钟线盒 5 分钟线制定规则上不同。分钟线显示的是后面一分钟的数据，也即 11:29 分钟线对应 11:29:00~11:29:59 数据，因此就没有 11:30 分钟线。5 分钟线对应的是前面 5 分钟的数据，即 9:35 对应 9:30:00~9:34:59 数据，并且集合竞价结果也被包含在 9:35 对应的 5 分钟线里。

Q: 怎么判断证券是否正在交易?

使用 WSQ 的 rt_susp_flag 字段

w.wsq('600058.SH','rt_susp_flag'), 结果是 5 位数字; 前 4 位为月份和日期, 停牌标志是五位整数
前四位是月份和日期, 最后一位含义如下:
=0 不停;=1 停 1h;=2 停 2h;=3 停半天;=4 停下午;=5 半小时;=6 临时停牌;=9 停牌一天
比如 7249, 表示 7 月 24 日停牌一天。

或者可以使用 wsd, wss 的 trade_status 字段判断。

5.3 交易接口查询返回的数据字段

5.3.1 资金查询返回消息

变量名-----是否必填项-----说明

股票返回应答-----

LogonID	Y	登录 LogonID
ErrID	Y	错误代号
ErrMsg	N	错误信息

Customer	Y	客户号
AssetAccount	Y	资金账号
DepartmentID	N	营业部 ID
MoneyType	Y	币种类型
Remark	N	说明
AvailableFund	Y	资金可用
BalanceFund	Y	资金余额
SecurityValue	N	持仓市值资产
FundAsset	N	资金资产
TotalAsset	N	总资产
Profit	N	总盈亏
FundFrozen	N	冻结资金
OtherFund	N	其他资金
BuyFund	N	今日买入金额
SellFund	N	今日卖出金额
期货返回应答-----		
LogonID	Y	登录 LogonID
ErrID	Y	错误代号
ErrMsg	N	错误信息
MoneyType	Y	币种类型
Customer	Y	客户号
AssetAccount	Y	资金账号
BalanceFund	Y	资金余额
AvailableFund	Y	资金可用
Remark	N	说明
FetchFund	Y	可取资金
ExerciseMargin	Y	履约保证金
RealFrozenMarginA	N	当日开仓预冻结金额
RealFrozenMarginB	N	当日开仓预冻结保证金和费用
HoldingProfit	N	盯市盈亏
TotalFloatProfit	N	总浮动盈亏
InitRightsBalance	N	期初客户权益
CurrRightsBalance	N	客户权益
FloatRightsBal	N	浮动客户权益
RealDrop	N	盯市平仓盈亏
RealDrop_Float	N	浮动平仓盈亏
FrozenFare	N	冻结费用
CustomerMargin	Y	客户保证金
RealOpenProfit	N	盯市开仓盈亏
FloatOpenProfit	N	浮动开仓盈亏
Interest	N	预计利息

5.3.2 持仓查询返回消息

变量名	是否必填项	说明
股票返回应答		
LogonID	Y	登录 LogonID
ErrID	Y	错误代号
ErrMsg	N	错误信息
Customer	Y	客户号
AssetAccount	Y	资金账号
MarketType	Y	证券市场
Shareholder	Y	股东代码
SecurityCode	Y	交易代码（证券代码/期货合约代码）
SecurityName	Y	交易品种名称（证券名称/期货合约名称）
DepartmentID	N	所属营业部
MoneyType	N	币种类型
Remark	N	说明
SecurityBalance	Y	股份余额
SecurityAvail	Y	股份可用
SecurityForzen	Y	股份冻结
TodayBuyVolume	N	当日买入数
TodaySellVolume	N	当日卖出数
SecurityVolume	N	当前持股数
CallVolume	N	可申赎数量
CostPrice	Y	成本价格
TradingCost	N	当前成本
LastPrice	N	最新价格
HoldingValue	N	市值
Profit	N	盈亏
期货返回应答		
LogonID	Y	登录 LogonID
ErrID	Y	错误代号
ErrMsg	N	错误信息
Customer	Y	客户号
AssetAccount	Y	资金账号
Shareholder	Y	期货账号-同股东代码
DepartmentID	N	所属营业部
SecurityCode	Y	交易代码（证券代码/期货合约代码）
SecurityName	Y	交易品种名称（证券名称/期货合约名称）
MarketType	Y	市场代码

MoneyType	N	币种类型
CostPrice	Y	成本价格
LastPrice	N	最新价格
TradeSide	Y	交易方向
BeginVolume	N	期初数量
EnableVolume	Y	可用数量
TodayRealVolume	N	当日可平仓数量
TodayOpenVolume	N	当日开仓可用数量
HoldingProfit	N	盯市盈亏
TotalFloatProfit	N	持仓浮动盈亏
PreMargin	N	上交易日保证金

5.3.3 当日委托查询返回消息

变量名	是否必填项	说明
股票返回应答		
LogonID	Y	登录 LogonID
ErrID	Y	错误代号
ErrMsg	N	错误信息
Customer	Y	客户号
AssetAccount	Y	资金账号
MoneyType	N	币种类型
MarketType	Y	证券市场
TradeSide	Y	交易方向
OrderType	N	"扩展标志- 价格委托方式
ExtFlag1	N	扩展标志
ExtFlag2	N	扩展标志
ExtFlag3	N	扩展标志
OrderStatus	Y	委托状态 包含Normal(正常)、Cancelled (撤单)、Invalid(无效)、Dealing(处理中)
Shareholder	N	股东代码
SecurityCode	Y	交易代码(证券代码/期货合约代码)
SecurityName	Y	交易品种名称(证券名称/期货合约 名称)
DepartmentID	N	所属营业部
OrderDate	Y	委托日期
OrderTime	Y	委托时间
OrderVolume	Y	委托数量
OrderPrice	Y	委托价格
TradedVolume	Y	成交数量
TradedPrice	N	成交均价

CancelVolume	Y	撤单数量
LastPrice	N	最新价格
OrderNumber	Y	柜台委托编号
Remark	N	说明
Seat	N	席位号
Agent	N	代理商号
OrderFrozenFund	N	委托冻结金额
MadeAmt	N	成交金额
期货返回应答-----		
LogonID	Y	登录 LogonID
ErrID	Y	错误代号
ErrMsg	N	错误信息
Customer	Y	客户号
AssetAccount	Y	资金账号
OrderDate	Y	委托日期
OrderTime	Y	委托时间
OrderVolume	Y	委托数量
OrderPrice	Y	委托价格
TradedVolume	Y	成交数量
TradedPrice	N	成交均价
CancelVolume	Y	撤单数量
LastPrice	N	最新价格
MarketType	Y	市场代码
OrderStatus	Y	委托状态 包含 Normal(正常)、Cancelled (撤单)、Invalid(无效)、Dealing(处理中)
Shareholder	Y	期货账号-同股东代码
SecurityCode	Y	交易代码(证券代码/期货合约代码)
SecurityName	Y	交易品种名称(证券名称/期货合约 名称)
OrderNumber	Y	柜台委托编号
Remark	N	说明
ExtFlag2	N	扩展标志
ExtFlag3	N	扩展标志
PreMargin	Y	开仓冻结保证金
TotalFrozenCosts	Y	冻结总费用
TradeSide	Y	交易方向
HedgeType	Y	套保标志
Seat	N	席位号
Agent	N	代理商号
Remark1	N	说明 1
Remark2	N	说明 2

5.3.4 当日成交查询返回消息

变量名	是否必填项	说明
股票返回应答		
LogonID	Y	登录 LogonID
ErrID	Y	错误代号
ErrMsg	N	错误信息
Customer	Y	客户号
AssetAccount	Y	资金账号
MoneyType	N	币种类型
MarketType	Y	证券市场
TradeSide	Y	交易方向
ExtFlag1	N	扩展标志
ExtFlag2	N	扩展标志
ExtFlag3	N	扩展标志
TradedStatus	Y	成交状态 包含 Normal (正常)、Cancelled (撤单)、Invalid (无效)
Shareholder	N	股东代码
SecurityCode	Y	交易代码 (证券代码/期货合约代码)
SecurityName	Y	交易品种名称 (证券名称/期货合约名称)
OrderDate	N	委托日期
OrderTime	N	委托时间
OrderVolume	N	委托数量
OrderPrice	N	委托价格
TradedVolume	Y	成交数量
TradedPrice	Y	成交价格
CancelVolume	N	撤单数量
TradedDate	Y	成交日期
TradedTime	Y	成交时间
LastPrice	N	最新价格
OrderNumber	Y	柜台委托编号
TradedNumber	Y	成交编号
Remark	N	说明
Remark1	N	其它说明
MadeAmt	Y	成交金额
期货返回应答		
LogonID	Y	登录 LogonID
ErrID	Y	错误代号
ErrMsg	N	错误信息
Customer	Y	客户号
AssetAccount	Y	资金账号

TradedDate	Y	成交日期
TradedTime	Y	成交时间
OrderVolume	N	委托数量
OrderPrice	N	委托价格
TradedVolume	Y	成交数量
TradedPrice	Y	成交价格
CancelVolume	N	撤单数量
LastPrice	N	最新价格
MarketType	Y	市场代码
ExtFlag1	N	扩展标志
ExtFlag2	N	扩展标志
ExtFlag3	N	扩展标志
TradedStatus	Y	成交状态包含 Normal(正常)、Cancelled(撤单)、Invalid(无效)
Shareholder	Y	期货账号-同股东代码
SecurityCode	Y	交易代码(证券代码/期货合约代码)
SecurityName	Y	交易品种名称(证券名称/期货合约名称)
OrderNumber	Y	柜台委托编号
TradedNumber	Y	成交编号
Remark	N	说明
Remark1	N	其它说明
AmountPerHand	N	每手吨数
TradeSide	Y	交易方向
HedgeType	Y	套保标志
TotalFrozenCosts	N	冻结总费用
DropProfit	N	平仓盈亏
DropFloatFrofit	N	平仓浮动盈亏
Seat	N	席位号
Agent	N	代理商号

5.3.5 营业部查询返回消息

 变量名-----是否必填项-----说明

ErrID	Y	错误代号
ErrMsg	N	错误信息
DepartmentID	Y	营业部 ID
DepartmentName	Y	营业部名称
AvailMarketFlag	N	可操作市场标识, 按位运算
LogonType	N	"可登录标识, 按位运算"

5.3.6 股东查询返回消息

变量名	是否必填项	说明

LogonID	Y	登录 LogonID
ErrID	Y	错误代号
ErrMsg	N	错误信息
ShareholderStatus	N	状态 (股东状态)
MainShareholderFlag	N	主股东标志
AccountType	Y	账号类型
MarketType	Y	市场代码
DepartmentID	Y	所属营业部
Shareholder	N	股东代码
CustomerName	N	客户姓名
AssetAccount	Y	资金账号
Customer	Y	客户号
Seat	N	席位号

5.3.7 券商（期货商）信息返回

变量名	是否必填项	说明

ErrID	Y	错误代号
ErrMsg	N	错误信息
BrokerID	Y	Broker 代号
BrokerName	Y	Broker 名称
ConnectModel	Y	连接模式 #0:连接 wts 1:直连 ctp 真 实环境 2:直连 ctp 模拟环境

5.3.8 已登录账户信息返回

变量名	是否必填项	说明

ErrID	Y	错误代号
ErrMsg	N	错误信息
LogonID	Y	登录 Logon 返回的 ID 代号
LogonAccount	Y	登录账号

AccountType

Y

账号类型