



# 会员端买断式回购单券市场计算接口 开发者指南

文 档 状 态	<input checked="" type="checkbox"/> 初稿	文档标识	
	<input type="checkbox"/> 评审通过		
	<input type="checkbox"/> 修改		
	<input type="checkbox"/> 发布	当前版本	1.0
	<input type="checkbox"/> 作废		

中国外汇交易中心暨全国银行间同业拆借中心

2023 年 12 月

变更履历

版本	更改时间	更改的图标和章节号	状态	更改简要描述	更改人	批准人
1.0	2024-2-1	全文	N	初稿		

注：状态可以为 N-新建、A-增加、M-更改、D-删除

## 目录

概述 .....	4
1 文档描述 .....	4
2 组件包版本 .....	4
适用场景 .....	5
使用详解 .....	6
1 会员端计算接口介绍 .....	6
1.1 到期结算日/实际占款天数/交易品种计算 .....	6
1.2 回购利率/首次净价/到期净价/应计利息计算 .....	7
1.3 首次结算金额/到期结算金额计算 .....	11

## 概述

### 1 文档描述

本文档适用于系统设计与开发人员，用于指导系统设计与开发人员使用计算接口计算首次结算日、到期结算日、实际占款天数、交易品种、回购利率、首次净价、首次收益率、到期净价、到期收益率、首次结算金额、到期结算金额、应计利息等要素。

### 2 组件包版本

tbs-rdi-interface-1.191.11-RELEASE.jar

## 适用场景

接口适用于买断式回购单券市场。

## 使用详解

### 1 会员端计算接口介绍

#### 1.1 到期结算日/实际占款天数/交易品种计算

##### 1.1.1 描述

输入清算速度、回购期限，计算首次结算日、到期结算日、实际占款天数、交易品种。

##### 1.1.2 接口对应 Java 类

OrpoCalculationRuleInterface(com.cfets.tbs.rdi.orpo)

##### 1.1.3 方法

getOrpoCalculation(ImixSession imixSession,ORPOCalculationInputValue  
inputValue,TimeOutParam... timeOutParams)

循环次数	字段类型	字段名	字段含义	是否必须	字段说明
输入参数					
imixSession					
1	ImixSession	ImixSession	session	必须	登录成功的 imixsession
ORPOCalculationInputValue					
1	Int	settlementSpeed	清算速度	必须	枚举值： 1-T+0 2-T+1 3-T+2 4-T+3 5-T+4
1	Int	repoMaturity	回购期限	必须	
输出参数					

1	Date	firstSettlement	首次结算日	必须	日期格式
1	Date	dueSettlement	到期结算日	必须	日期格式
1	Int	occupancyDays	实际占款天数	必须	整数
1	String	instrument	交易品种	必须	字符串

#### 1.1.4 示例

imixSession ;//登陆成功后的 imixSession

```
ORPOCalculationInputValue inputValue = new ORPOCalculationInputValue ();
```

```
inputValue.setSettlementSpeed(2);//清算速度
```

```
inputValue.setRepoMaturity(20);//回购期限
```

```
TimeOutParam timeOutParam = new TimeOutParam("NDM","10");//10 代表是买断单券
```

```
ORPOCalculationInputValue outputValue
```

```
OrpoCalculationRuleInterface.getOrpoCalculation(imixSession,inputValue,timeOutParam  
)
```

```
Date firstSettlement = outputValue .getFirstSettlement ();//首次结算日
```

```
Date dueSettlement= outputValue .getDueSettlement();//到期结算日
```

```
int occupancyDays= outputValue .getOccupancyDays();//实际占款天数
```

```
String instrument= outputValue .getInstrument();//交易品种
```

#### 1.1.5 注意事项

因网络可能存在延迟,TimeOutParam 属性是设置计算接口等待时间的一个参数,目前默认计算接口等待时间为 5 秒,5 秒之后如果后台没返回计算结果,则认为本次计算失败;如果会员想自定义超时时间,可以指定 new TimeOutParam("NDM","10"),此时默认时间会提高到 30 秒.

### 1.2 回购利率/首次净价/到期净价/应计利息计算

#### 1.2.1 描述

输清算速度、回购期限、债券代码、回购利率、首次净价（首次收益率）、到期净价（到期收益率），计算回购利率、首次净价（首次收益率）、到期净价（到期收益率）、首次结算日、到期结算日、首次应计利息、到期应计利息等要素。

其中，回购利率、首次净价（首次收益率）、到期净价（到期收益率）三组要素之间互相推算，即任意输入其中的两组要素，系统自动计算第三组；若三组要素均输入，则通过首次净价（首次收益率）和到期净价（到期收益率）推算回购利率。

### 1.2.2 接口对应 Java 类

OrpoCalculationRuleInterface(com.cfets.tbs.rdi.orpo)

### 1.2.3 方法

getOrpoCalculation(ImixSession imixSession,ORPOCalculationInputValue  
inputValue,TimeOutParam... timeOutParams)

循环次数	字段类型	字段名	字段含义	是否必须	字段说明
输入参数					
imixSession					
1	ImixSession	ImixSession	session	必须	登录成功的 imixsession
ORPOCalculationInputValue					
1	Int	settlementSpeed	清算速度	必须	枚举值： 1-T+0 2-T+1 3-T+2 4-T+3 5-T+4
1	Int	repoMaturity	回购期限	必须	单位：天； 正整数，大于0小于等于365
1	String	bondCode	债券代码	必须	
1	BigDecimal	repoRate	回购利率	非必须	
1	BigDecimal	firstNetPrc	首次净价	非必须	首次净价



					和首次收 益率传其 一即可，若 均传，以首 次净价为 准
1	BigDecimal	firstYield	首次收 益 率	非必须	首次净价 和首次收 益率传其 一即可，若 均传，以首 次净价为 准
1	BigDecimal	firstNetPrc	到期净价	非必须	到期净价 和到期收 益率传其 一即可，若 均传，以到 期净价为 准
1	BigDecimal	firstYield	到期收 益 率	非必须	
输出参数					
1	Int	occupancyDays	实际占款 天数	必须	整数
1	String	instrument	交易品种	必须	字符串,如 OR014
1	String	BondCode	债券代码	必须	
1	BigDecimal	firstNetPrc	首次净价	必须	
1	BigDecimal	firstYield	首次收 益 率	必须	
1	Date	firstSettlementDate	首次结算 日	必须	日期格式
1	BigDecimal	FirstAccrdIntrst	首次应计 利息	必须	
1	BigDecimal	dueNetPrice	到期净价	必须	

1	BigDecimal	dueYield	到期收益率	必须	
1	Date	dueSettlementDate	到期结算日	必须	日期格式
1	BigDecimal	dueAccrdIntrst	到期应计利息	必须	
1	BigDecimal	repoRate	回购利率	必须	

#### 1.2.4 示例

imixSession ;//登陆成功后的 imixSession

ORPOCalculationInputValue inputValue = new ORPOCalculationInputValue ();

inputValue.setBondCode("1805384");//债券代码

inputValue.setSettlementSpeed(2);//清算速度

inputValue.setRepoRate(new BigDecimal("3.2620"));//回购利率

inputValue.setRepoMaturity(20);//回购期限

inputValue.setFirstNetPrc(new BigDecimal("100"));//首次净价

TimeOutParam timeOutParam = new TimeOutParam("NDM","10");//10 代表是买断单券

ORPOCalculationInputValue outputValue

OrpoCalculationRuleInterface.getOrpoCalculation(imixSession,inputValue,timeOutParam  
)

int occupancyDays= outputValue .getOccupancyDays();//实际占款天数

String instrument= outputValue .getInstrument();//交易品种

BigDecimal firstNetPrc = outputValue .getFirstNetPrc();//首次净价

BigDecimal firstYield= outputValue .getFirstYield();//首次收益率

BigDecimal firstSettlementDate= outputValue .getFirstSettlementDate();//首次结算日

BigDecimal firstAccrdIntrst= outputValue .getFirstAccrdIntrst();//首次应计利息

BigDecimal dueNetPrice= outputValue .getDueNetPrice();//到期净价

BigDecimal dueYield= outputValue .getDueYield();//到期收益率

Date dueSettlementDate= outputValue .getDueSettlementDate();//到期结算日

BigDecimal dueAccrdIntrst= outputValue .getDueAccrdIntrst();//到期应计利息

```
BigDecimal repoRate= outputValue .getRepoRate();//回购利率
```

### 1.2.5 注意事项

当输入回购利率与首次净价（首次收益率），或回购利率与到期净价（到期收益率）时，回购利率可能会被重置。

## 1.3 首次结算金额/到期结算金额计算

### 1.3.1 描述

输入清算速度、回购期限、回购利率、债券代码、券面总额、首次净价（首次收益率）、到期净价（到期收益率）、回购利率，计算首次结算金额、到期结算金额等要素。

其中，回购利率、首次净价（首次收益率）、到期净价（到期收益率）三组要素之间互相推算，即任意输入其中的两组要素，系统自动计算第三组；若三组要素均输入，则通过首次净价（首次收益率）和到期净价（到期收益率）推算回购利率。

### 1.3.2 接口对应 Java 类

```
OrpoCalculationRuleInterface(com.cfets.tbs.rdi.orpo)
```

### 1.3.3 方法

```
getOrpoCalculation(ImixSession imixSession,ORPOCalculationInputValue  
inputValue,TimeOutParam... timeOutParams)
```

循环次数	字段类型	字段名	字段含义	是否必须	字段说明
输入参数					
imixSession					
1	ImixSession	ImixSession	session	必须	登录成功的 imixsession
ORPOCalculationInputValue					
1	Int	settlementSpeed	清算速度	必须	枚举值： 1-T+0 2-T+1 3-T+2

会员端买断式回购单券市场计算接口开发者指引

					4-T+3 5-T+4
1	Int	repoMaturity	回购期限	必须	大于 0 小于 等于 365
1	String	bondCode	债券代码	必须	
1	BigDecimal	firstNetPrc	首次净价	非必须	首次净价 和首次收 益率传其 一即可，若 均传，以首 次净价为 准
1	BigDecimal	firstYield	首次收益 率	非必须	首次净价 和首次收 益率传其 一即可，若 均传，以首 次净价为 准
1	BigDecimal	dueNetPrice	到期净价	非必须	到期净价 和到期收 益率传其 一即可，若 均传，以到 期净价为 准
1	BigDecimal	dueYield	到期收益 率	非必须	到期净价 和到期收 益率传其 一即可，若 均传，以到 期净价为 准
1	BigDecimal	amnt	券面总额	必须	
1	BigDecimal	repoRate	回购利率	非必须	

输出参数					
1	Int	occupancyDays	实际占款天数	必须	整数
1	String	instrument	交易品种	必须	字符串,如OR014
1	String	BondCode	债券代码	必须	
1	BigDecimal	firstNetPrc	首次净价	必须	
1	BigDecimal	firstYield	首次收益率	必须	
1	Date	firstSettlementDate	首次结算日	必须	日期格式
1	BigDecimal	FirstAccrdIntrst	首次应计利息	必须	
1	BigDecimal	dueNetPrice	到期净价	必须	
1	BigDecimal	dueYield	到期收益率	必须	
1	Date	dueSettlementDate	到期结算日	必须	日期格式
1	BigDecimal	dueAccrdIntrst	到期应计利息	必须	
1	BigDecimal	repoRate	回购利率	必须	
1	BigDecimal	firstSettlementAmnt	首次结算金额	必须	
1	BigDecimal	dueSettlementAmnt	到期结算金额	必须	

### 1.3.4 示例

imixSession ;//登陆成功后的 imixSession

ORPOCalculationInputValue inputValue = new ORPOCalculationInputValue ();

inputValue.setBondCode("1805384");//债券代码

inputValue.setSettlementSpeed(2);//清算速度

```
inputValue.setRepoRate(new BigDecimal("3.2620")); //回购利率
inputValue.setRepoMaturity(20); //回购期限
inputValue.setFirstNetPrc(new BigDecimal("100")); //首次净价
inputValue.setDueNetPrc(new BigDecimal("99.9835")); //到期净价
inputValue.setAmnt(new BigDecimal("10000000")); //券面总额

TimeOutParam timeOutParam = new TimeOutParam("NDM", "10"); //10 代表是买断单券
ORPOCalculationInputValue outputValue
OrpoCalculationRuleInterface.getOrpoCalculation(imixSession, inputValue, timeOutParam
)
int occupancyDays= outputValue .getOccupancyDays(); //实际占款天数
String instrument= outputValue .getInstrument(); //交易品种
BigDecimal firstNetPrc= outputValue .getFirstNetPrc(); //首次净价
BigDecimal firstYield= outputValue .getFirstYield(); //首次收益率
BigDecimal firstSettlementDate= outputValue .getFirstSettlementDate(); //首次结算日
BigDecimal firstAccrdIntrst= outputValue .getFirstAccrdIntrst(); //首次应计利息
BigDecimal dueNetPrice= outputValue .getDueNetPrice(); //到期净价
BigDecimal dueYield= outputValue .getDueYield(); //到期收益率
Date dueSettlementDate= outputValue .getDueSettlementDate(); //到期结算日
BigDecimal dueAccrdIntrst= outputValue .getDueAccrdIntrst(); //到期应计利息
BigDecimal repoRate= outputValue .getRepoRate(); //回购利率
BigDecimal firstSettlementAmnt= outputValue .getFirstSettlementAmnt(); //首次结算金
额
BigDecimal dueYield= outputValue .getDueSettlementAmnt(); //到期结算金额
```