



# **Клиентская библиотека расчета обеспечения срочного рынка**

**Описание API**

**Версия 8.0**

**Москва 2025**



---

# Содержание

1. Назначение библиотеки .....	12
2. Архитектура библиотеки .....	13
2.1. Типы конфигурации библиотеки .....	13
2.2. Состав дистрибутива .....	13
2.3. Принципиальная схема работы модуля SpectralM .....	13
3. Описание API .....	16
3.1. Общий перечень функций .....	16
3.2. Сервисные функции .....	18
3.2.1. GetSpectralMLibraryVersion .....	18
3.2.2. GetErrorCodeDescription .....	18
3.3. Функции инициализации и создания модуля расчета .....	18
3.3.1. InitSpectralM .....	18
3.3.2. DelInitSpectralM .....	19
3.3.3. CreateSpectralMCalculator .....	19
3.3.4. DestroySpectralMCalculator .....	19
3.4. Функции связи с сервером .....	19
3.4.1. GetInstrumentParameters .....	19
3.5. Функции для получения параметров инструментов из расчетчика .....	20
3.5.1. SetOnFirstCurrency .....	20
3.5.2. SetOnFirstBaseContract .....	20
3.5.3. SetOnFirstFutures .....	21
3.5.4. SetOnFirstOptionSeries .....	21
3.5.5. SetOnFirstOption .....	21
3.5.6. SetOnFirstMultileg .....	22
3.5.7. GetCommonParams .....	22
3.5.8. GetNextCurrency .....	22
3.5.9. GetNextBaseContract .....	22
3.5.10. GetNextFutures .....	24
3.5.11. GetNextOptionSeries .....	26
3.5.12. GetNextOption .....	29
3.5.13. GetNextMultileg .....	29
3.6. Функции для заливки инструментов в модуль расчетчика .....	30
3.6.1. SetCommonParams .....	30
3.6.2. InsertCurrency .....	30
3.6.3. InsertBaseContract .....	30
3.6.4. InsertFutures .....	32
3.6.5. InsertOptionSeries .....	33
3.6.6. InsertOption .....	36
3.6.7. InsertMultileg .....	36
3.6.8. InsertBrokerBaseContractParam .....	37
3.7. Функции для управления данными о фирмах и клиентах .....	37
3.7.1. RegisterSettlementAccount .....	37
3.7.2. RegisterFirm .....	38
3.7.3. RegisterPart .....	39
3.7.4. ChangePart .....	40
3.8. Функции для управления позициями и заявками .....	41
3.8.1. SetPosition .....	41
3.8.2. ChangeOrder .....	42
3.9. Функции очистки .....	43
3.9.1. ClearParts .....	43
3.9.2. ClearAll .....	43
3.10. Функции для получения массивов рисков клиентов .....	43
3.10.1. GetClientVfRisksArray .....	43
3.10.2. GetClientDeliveryRisksArray .....	44
3.10.3. GetClientFutRisksArray .....	45
3.10.4. GetClientMmsRisksArray .....	46
3.10.5. GetClientMcsRisksArray .....	47
3.10.6. GetBrokerRisksArray .....	47
3.10.7. GetRiskPoints .....	48
3.10.8. GetClientFinalRisksArray .....	49
3.10.9. GetBrokerFinalRisksArray .....	50
3.10.10. GetSettlementAccountFinalRisksArray .....	51
3.10.11. GetClientFinalRisksArrayWithPositiveValue .....	51
3.10.12. GetBrokerFinalRisksArrayWithPositiveValue .....	52
3.10.13. GetSettlementAccountFinalRisksArrayWithPositiveValue .....	52
3.10.14. Структура ExtraRisksInfo .....	53
3.11. Возвращаемые коды ошибок .....	54

# История изменений

Дата	Версия	Изменения
28.02.2025	8.0	<p>1. В функции <b>GetNextBaseContract</b> и <b>InsertBaseContract</b> добавлены новые параметры:</p> <ul style="list-style-type: none"> <li>• <b>MarketRiskLow</b> - Ставка рыночного риска для начального уровня риска клиента</li> <li>• <b>MarketRisk1Low</b> - Ставка рыночного риска первого лимита концентрации для начального уровня риска клиента</li> <li>• <b>MarketRisk2Low</b> - Ставка рыночного риска второго лимита концентрации для начального уровня риска клиента</li> <li>• <b>MarketRiskMedium</b> - Ставка рыночного риска для стандартного уровня риска клиента</li> <li>• <b>MarketRisk1Medium</b> - Ставка рыночного риска первого лимита концентрации для стандартного уровня риска клиента</li> <li>• <b>MarketRisk2Medium</b> - Ставка рыночного риска второго лимита концентрации для стандартного уровня риска клиента</li> <li>• <b>MarketRiskHigh</b> - Ставка рыночного риска для повышенного уровня риска клиента</li> <li>• <b>MarketRisk1High</b> - Ставка рыночного риска первого лимита концентрации для повышенного уровня риска клиента</li> <li>• <b>MarketRisk2High</b> - Ставка рыночного риска второго лимита концентрации для повышенного уровня риска клиента</li> <li>• <b>MarketRiskExceptional</b> - Ставка рыночного риска для особого уровня риска клиента</li> <li>• <b>MarketRisk1Exceptional</b> - Ставка рыночного риска первого лимита концентрации для особого уровня риска клиента</li> <li>• <b>MarketRisk2Exceptional</b> - Ставка рыночного риска второго лимита концентрации для особого уровня риска клиента</li> </ul> <p>2. В функции <b>RegisterPart</b> и <b>GetNextFutures</b> добавлен новый параметр:</p> <ul style="list-style-type: none"> <li>• <b>ClientRiskLevel</b> - Категория уровня риска клиента <ul style="list-style-type: none"> <li>• <b>0</b> - Клиент с базовым уровнем риска (значение по умолчанию)</li> <li>• <b>1</b> - Клиент с Начальным Уровнем Риска (КНУР)</li> <li>• <b>2</b> - Клиент со Стандартным Уровнем Риска (КСУР)</li> <li>• <b>3</b> - Клиент с Повышенным Уровнем Риска (КПУР)</li> <li>• <b>4</b> - Клиент с Особым Уровнем Риска (КОУР)</li> </ul> </li> </ul> <p>3. В функцию <b>GetRiskPoints</b> добавлен новый параметр:</p> <ul style="list-style-type: none"> <li>• <b>level</b> - Категория уровня риска клиента <ul style="list-style-type: none"> <li>• <b>0</b> - Клиент с базовым уровнем риска (значение по умолчанию)</li> <li>• <b>1</b> - Клиент с Начальным Уровнем Риска (КНУР)</li> <li>• <b>2</b> - Клиент со Стандартным Уровнем Риска (КСУР)</li> <li>• <b>3</b> - Клиент с Повышенным Уровнем Риска (КПУР)</li> <li>• <b>4</b> - Клиент с Особым Уровнем Риска (КОУР)</li> </ul> </li> </ul> <p>4. В функции <b>RegisterPart</b> и <b>InsertBrokerBaseContractParam</b> добавлен новый параметр:</p> <ul style="list-style-type: none"> <li>• <b>GORatio</b> - Коэффициент клиентского ГО</li> </ul>
17.09.2024	7.30	1. В состав дистрибутива включена утилита для смены пароля - <b>change_password</b> .
22.01.2024	7.21	В текущей версии TAS фьючерсы недоступны.

Дата	Версия	Изменения
		<p>Внесённые изменения:</p> <ol style="list-style-type: none"> <li>В функцию <b>InsertFutures</b> добавлены: <ul style="list-style-type: none"> <li>Новый параметр <b>TASBaseFutCode</b> - Код опорного фьючерса.</li> <li>Новое значение параметра Attribute: <b>0x100000</b> - TAS фьючерс.</li> </ul> </li> <li>В функцию <b>GetNextFutures</b> добавлены: <ul style="list-style-type: none"> <li>Новый параметр <b>TASBaseFutCode</b> - Код опорного фьючерса.</li> <li>Новый параметр <b>TASBaseFutCodeSize</b> - Размер буфера под код опорного фьючерса, указанный в <b>TASBaseFutCode</b>.</li> <li>Новое значение параметра Attribute: <b>0x100000</b> - TAS фьючерс.</li> </ul> </li> <li>В функцию <b>InsertMultileg</b> добавлен новый код возврата при повторной вставке мультилега: <ul style="list-style-type: none"> <li><b>94792</b> - мультилег уже существует</li> </ul> </li> </ol>
06.09.2023	7.18	<ol style="list-style-type: none"> <li>В связи с введением опционов на индексы для параметра появляется дополнительный смысл, что отражено в описании: <ul style="list-style-type: none"> <li><b>r2</b> - Безрисковая ставка FX2 валютной пары FX2/FX1 (для премиальных опционов на валюту); ставка дивидендной доходности q (для премиальных опционов на индекс).</li> </ul> </li> </ol>
05.07.2023	7.15	<ol style="list-style-type: none"> <li>В функции <b>InsertOptionSeries</b> и <b>GetNextOptionSeries</b> добавлены новые параметры: <ul style="list-style-type: none"> <li><b>r2</b> - Безрисковая процентная ставка для валюты FX2 в валютной паре вида FX2/FX1</li> <li><b>InterestRate2RiskUp</b> - Значение ставки рассогласования процентного риска для ставки r2 (сценарий движения ставки вверх)</li> <li><b>InterestRate2RiskDown</b> - Значение ставки рассогласования процентного риска для ставки r2 (сценарий движения ставки вниз)</li> </ul> </li> </ol>
20.04.2023	7.12	<ol style="list-style-type: none"> <li>В функции <b>InsertBaseContract</b> параметр <b>ShortOptionMinimumChargeRatio</b> переименован в <b>ShortOptionMinimumCharge</b>.</li> <li>В функции <b>InsertMultileg</b> параметр <b>Code</b> переименован в <b>MultilegCode</b>.</li> <li>В функцию <b>GetNextBaseContract</b> добавлены новые параметры: <ul style="list-style-type: none"> <li><b>MarketRisk</b> - Ставка рыночного риска</li> <li><b>MarketRisk1</b> - Ставка рыночного риска для первого лимита концентрации</li> <li><b>MarketRisk2</b> - Ставка рыночного риска для второго лимита концентрации</li> <li><b>ConcentrationLimit1</b> - Первый лимит концентрации позиции</li> <li><b>ConcentrationLimit2</b> - Второй лимит концентрации позиции</li> </ul> </li> <li>В функцию <b>GetNextFutures</b> добавлены новые параметры: <ul style="list-style-type: none"> <li><b>InterestRateRiskUp</b> - Значение ставки рассогласования процентного риска (сценарий движения вверх)</li> <li><b>InterestRateRiskDown</b> - Значение ставки рассогласования процентного риска (сценарий движения вниз)</li> <li><b>TimeToExpiration</b> - Значение времени до экспирации фьючерса в долях года</li> <li><b>NormalizedSpot</b> - Риск-параметр, означающий рассчитанную цену спота для данного базового контракта, нормализованный к размерности фьючерса</li> </ul> </li> <li>В функции <b>GetNextOptionSeries</b> добавлены новые параметры: <ul style="list-style-type: none"> <li><b>VolatilityRisk</b> - Ставка риска волатильности</li> <li><b>VolatilityRiskMismatch</b> - Ставка рассогласования риска волатильности</li> </ul> </li> </ol>

Дата	Версия	Изменения
		<ul style="list-style-type: none"> <li>• <b>TimeToExpiration</b> - Значение времени до экспирации опционной серии в долях года</li> <li>• для параметра <b>UseNullVolat</b> изменена размерность с int32 на int8.</li> </ul>
23.01.2023	7.9	<ol style="list-style-type: none"> <li>1. Из функции <b>InsertBaseContract</b> удален параметр <b>Type</b> (Признак процентного контракта).</li> <li>2. Добавлена новая функция <b>InsertBrokerBaseContractParam</b> - Добавление коэффициента ГО в разрезе брокера и базового актива.</li> </ol>
28.09.2022	7.6	<ol style="list-style-type: none"> <li>1. Из функции <b>RegisterSettlementAccount</b> удален параметр <b>SettlementAccountType</b> (Тип РК), так как он не влияет на расчет ГО.</li> <li>2. Из функций <b>RegisterPart</b> и <b>ChangePart</b> удален параметр <b>BrokerCode</b> (Код БФ, которой принадлежит клиент).</li> <li>3. Из функции <b>ChangeOrder</b> удален параметр <b>NoteUsed</b>.</li> </ol>
01.03.2022	7.3	<ol style="list-style-type: none"> <li>1. Добавлены новые функции: <ul style="list-style-type: none"> <li>• <b>GetErrorCodeDescription</b> - Получение описания ошибки по коду возврата</li> <li>• <b>GetCommonParams</b> - Получение общих параметров</li> <li>• <b>SetOnFirstCurrency</b> - Функция для установки начала итерации списка валют</li> <li>• <b>GetNextCurrency</b> - Итерация списка валют</li> </ul> </li> <li>2. Переименованы следующие функции: <ul style="list-style-type: none"> <li>• <b>GetSpectralMVersion</b> в <b>GetSpectralMLibraryVersion</b></li> <li>• <b>InsertVirtFutures</b> в <b>InsertOptionSeries</b></li> <li>• <b>SetOnFirstVirtFutures</b> в <b>SetOnFirstOptionSeries</b></li> <li>• <b>GetNextVirtFutures</b> в <b>GetNextOptionSeries</b></li> </ul> </li> <li>3. Изменение параметров функций: <ul style="list-style-type: none"> <li>• из функции <b>InsertBaseContract</b> удален параметр <b>Type</b> (вместо него следует ориентироваться на <b>AssetClass</b>)</li> <li>• из функции <b>InsertCurrency</b> удален параметр <b>Signs</b>, ранее объявленный устаревшим</li> <li>• в функции <b>GetNextBaseContract</b>, <b>GetNextFutures</b>, <b>GetNextOptionSeries</b>, <b>GetNextOption</b>, <b>GetNextMultileg</b> добавлены параметры для передачи размера буфера под параметры инструментов определенного типа: например, <b>FuturesCodeSize</b> или <b>BaseContractCodeSize</b>. У этих функций для каждого строкового параметра добавлен дополнительно размер, таким образом передается буфер и размер, а функция возвращает, сколько она туда записала.</li> <li>• переименованы параметры в следующих функциях: <ul style="list-style-type: none"> <li>• в <b>InitSpectralM</b> - <b>IniFile</b> в <b>IniFilePath</b></li> <li>• в <b>InsertCurrency</b> - <b>IndicatorID</b> в <b>CurrencyId</b></li> <li>• в <b>InsertBaseContract</b> и <b>GetNextBaseContract</b> - <b>Code</b> в <b>BaseContractCode</b>, <b>CodeMCS</b> в <b>MCSCode</b>, <b>IndicatorID</b> в <b>CurrencyId</b>, <b>SOMC</b> в <b>ShortOptionMinimumChargeRatio</b></li> <li>• в <b>InsertFutures</b> и <b>GetNextFutures</b> - <b>Code</b> в <b>FuturesCode</b>, <b>CodeBC</b> в <b>BaseContractCode</b></li> <li>• в <b>InsertOptionSeries</b> и <b>GetNextOptionSeries</b> - <b>Code</b> в <b>OptionSeriesCode</b>, <b>CodeRF</b> в <b>FuturesCode</b></li> <li>• в <b>InsertOption</b> и <b>GetNextOption</b> - <b>Code</b> в <b>OptionCode</b>, <b>CodeVF</b> в <b>OptionSeriesCode</b></li> <li>• в <b>RegisterFirm</b> - <b>Code</b> в <b>BrokerCode</b></li> <li>• в <b>RegisterPart</b> и <b>ChangePart</b> - <b>Code</b> в <b>ClientCode</b>, <b>FirmCode</b> в <b>BrokerCode</b></li> </ul> </li> </ul> </li> </ol>

Дата	Версия	Изменения
		<ul style="list-style-type: none"> <li>• в <b>SetPosition - RecordType</b> в AccountHierarchyRecordType</li> <li>• в <b>SetOnFirstOptionSeries - nVirtFutures</b> в OptionSeriesCount</li> <li>• в <b>SetOnFirstOption - nOption</b> в OptionCount</li> <li>• в <b>GetNextMultileg - Code</b> в MultilegCode</li> </ul> <p>4. Из публичного интерфейса удалены следующие функции, рекомендуется использовать аналогичные функции без окончания Extra:</p> <ul style="list-style-type: none"> <li>• SetOnFirstCurrencyExtra</li> <li>• SetOnFirstBaseContractExtra</li> <li>• SetOnFirstFuturesExtra</li> <li>• SetOnFirstVirtFuturesExtra</li> <li>• SetOnFirstOptionExtra</li> <li>• GetCommonParametersExtra</li> <li>• GetNextCurrencyExtra</li> <li>• GetNextBaseContractExtra</li> <li>• GetNextFuturesExtra</li> <li>• GetNextVirtFuturesExtra</li> <li>• GetNextOptionExtra</li> <li>• GetNextMultilegExtra</li> </ul> <p>5. Изменены типы данных:</p> <ul style="list-style-type: none"> <li>• добавлены размерности целочисленных типов (int → int32_t/int8_t)</li> <li>• добавлены перечисления с актуальными значениями вместо целочисленных типов для параметров: <ul style="list-style-type: none"> <li>• MSPTYPE</li> <li>• OptionModel</li> <li>• AssetClass</li> <li>• SpreadAspect</li> <li>• MarginStyle</li> <li>• SettlementType</li> <li>• ExerciseStyle</li> <li>• OptionType</li> <li>• SettlementAccountType</li> <li>• CalendarSpreadMarginType</li> <li>• InterContractSpreadMarginType</li> <li>• BrokerAccountType</li> <li>• MarginRule</li> <li>• accountHierarchy</li> </ul> </li> </ul> <p>6. Новый код возврата:</p> <ul style="list-style-type: none"> <li>• <b>94995</b> - Повторный вызов функции GetInstrumentParameters без предварительного вызова функции очистки ClearAll</li> </ul>

Дата	Версия	Изменения
22.12.2021	7.0	<ol style="list-style-type: none"> <li>Из функций <b>GetNextBaseContract</b> и <b>InsertBaseContract</b> удален параметр <b>SubRiskStep</b>.</li> <li>В функции <b>GetNextBaseContract</b> и <b>InsertBaseContract</b> добавлены следующие параметры: <ul style="list-style-type: none"> <li>• <b>AssetClass</b> - Классификация типов базового актива</li> <li>• <b>CFRisk</b> - Ставка риска изменения размера прогнозного денежного потока</li> </ul> </li> <li>Из функций <b>GetNextFutures</b> и <b>InsertFutures</b> удален параметр <b>SubRisk</b>.</li> <li>В функции <b>GetNextVirtFutures</b> и <b>InsertVirtFutures</b> добавлены следующие параметры: <ul style="list-style-type: none"> <li>• <b>MarginStyle</b> - Способ маржирования опциона: 0 - маржируемый 1 – премиальный</li> <li>• <b>SettlementType</b> - Тип опциона: 0 - расчетный 1 - поставочный</li> <li>• <b>ExerciseStyle</b> - Способ исполнения опциона: 0 - американский 1 - европейский</li> <li>• <b>SubRisk</b> - Признак учета риска по подточкам: 1-Включен режим учета риска по подточкам риска, 0 – Выключен</li> <li>• <b>SpreadAspec</b> - Флаг участия в спреде</li> <li>• <b>EnforceHalfNetting</b> - Признак принудительного разрыва MMC</li> <li>• <b>MinStep</b> - Минимальный шаг цены (в пунктах)</li> <li>• <b>MinStepPrice</b> - Цена минимального шага (в деньгах)</li> <li>• <b>LotCoeff</b> - Количество единиц базового актива в инструменте</li> <li>• <b>r</b> - Безрисковая процентная ставка</li> <li>• <b>InterestRateRiskUp</b> - Значение ставки рассогласования процентного риска (сценарий движения вверх).</li> <li>• <b>InterestRateRiskDown</b> - Значение ставки рассогласования процентного риска (сценарий движения вниз).</li> <li>• <b>FixedSpotDiscount</b> - Сумма дисконтированных значений объявленных денежных потоков</li> <li>• <b>ProjectedSpotDiscount</b> - Сумма дисконтированных значений прогнозных денежных потоков</li> </ul> </li> <li>В функцию <b>SetPosition</b> добавлен параметр <b>AfterLastClearingQuantity</b> - Объем позиции на момент последнего клиринга (ПК или ВК).</li> </ol>
12.10.2021	6.15	<ol style="list-style-type: none"> <li>Параметр <b>Signs</b> в функции <b>InsertCurrency</b> объявлен устаревшим и будет удален в версии 7.3.</li> </ol>
12.05.2021	6.9	<ol style="list-style-type: none"> <li>В функцию <b>InsertFutures</b> добавлен параметр <b>EnforceIMSHalfNetting</b>.</li> </ol>
10.08.2020	6.6	<ol style="list-style-type: none"> <li>Из функции <b>GetNextVirtFutures</b> удален параметр <b>FuturesType</b>.</li> </ol>
29.07.2020	6.5	<ol style="list-style-type: none"> <li>В функции <b>InsertBaseContract</b> и <b>GetNextBaseContract</b> добавлен параметр <b>SpotPrice</b>.</li> <li>В функции <b>InsertVirtFutures</b> и <b>GetNextVirtFutures</b> добавлен параметр <b>M_Bach</b> для расчета волатильности по формуле Башелье.</li> </ol>
03.06.2020	6.4.20	<ol style="list-style-type: none"> <li>Изменения в функции <b>GetNextBaseContract</b>: <ul style="list-style-type: none"> <li>• удалены параметры <b>HasOption</b></li> <li>• новый параметр <b>OptionModel</b> - Опционная модель на данном БА <ul style="list-style-type: none"> <li>• 0 - Black-Scholes</li> <li>• 1 - Bacheliers</li> </ul> </li> </ul> </li> <li>Изменения в функции <b>InsertBaseContract</b>: <ul style="list-style-type: none"> <li>• удалены параметры <b>SpotPrice</b> и <b>HasOption</b></li> </ul> </li> </ol>



Дата	Версия	Изменения
		<ul style="list-style-type: none"> <li>новый параметр <b>OptionModel</b> - Опционная модель на данном БА               <ul style="list-style-type: none"> <li>0 - Black-Scholes</li> <li>1 - Bacheliers</li> </ul> </li> </ul> <p>3. Изменения в функциях <b>InsertFutures</b> и <b>GetNextFutures</b>:</p> <ul style="list-style-type: none"> <li>новые параметры               <ul style="list-style-type: none"> <li><b>Attribute</b> - Признак спот-фьючерса</li> <li><b>MRAddOnUp</b> - Надбавка Up на рыночный риск отдельного фьючерса</li> <li><b>MRAddOnDown</b> - Надбавка Down на рыночный риск отдельного фьючерса</li> </ul> </li> </ul> <p>4. Изменения в функциях <b>InsertVirtFutures</b> и <b>GetNextVirtFutures</b>:</p> <ul style="list-style-type: none"> <li>переименованы параметры A, B, C, D, E, S соответственно в A_Black, B_Black, C_Black, D_Black, E_Black, S_Black - Массив параметров функции расчета волатильности по формуле Блэка Шоулза</li> <li>новые параметры A_Bach, B_Bach, C_Bach, D_Bach, E_Bach, S_Bach - Массив параметров функции расчета волатильности по формуле Башелье</li> </ul>
23.05.2019	6.2.20	<p>1. Добавлены новые параметры <b>ShortOptionMinimumChargeRatio</b> (индивидуальная процентная ставка) и <b>InterContractSpreadMarginType</b> (правило маржирования для контрактов, входящих в МКС) в функции:</p> <ul style="list-style-type: none"> <li><b>RegisterPart</b></li> <li><b>RegisterFirm</b></li> <li><b>RegisterSettlementAccount</b></li> </ul> <p>2. Добавлен новый параметр <b>WindowSizePercentage</b> (диапазон сглаживания в процентах от ставки риска) в следующие функции:</p> <ul style="list-style-type: none"> <li><b>InsertBaseContract</b></li> <li><b>GetNextBaseContract</b></li> </ul>
07.03.2018	6.0	<p>1. Добавлены новые функции для мультилегов:</p> <ul style="list-style-type: none"> <li><b>InsertMultileg</b></li> <li><b>GetNextMultilegExtra</b></li> <li><b>GetNextMultileg</b></li> </ul> <p>2. Добавлена новые функции возврата финальных риск-строк клиента и брокера:</p> <ul style="list-style-type: none"> <li><b>GetClientFinalRisksArray</b></li> <li><b>GetBrokerFinalRisksArray</b></li> </ul> <p>3. Изменен состав параметров в следующих функциях:</p> <ul style="list-style-type: none"> <li><b>GetNextVirtFutures</b></li> <li><b>InsertBaseContract</b></li> <li><b>InsertFutures</b></li> <li><b>InsertVirtFutures</b></li> <li><b>RegisterSettlementAccount</b></li> <li><b>RegisterFirm</b></li> <li><b>RegisterPart</b></li> <li><b>SetPosition</b></li> <li><b>ChangeOrder</b></li> </ul>

Дата	Версия	Изменения
		<ul style="list-style-type: none"> <li>• <b>GetClientVfRisksArray</b></li> <li>• <b>GetClientDeliveryRisksArray</b></li> <li>• <b>GetClientFutRisksArray</b></li> <li>• <b>GetClientMmsRisksArray</b></li> <li>• <b>GetClientMcsRisksArray</b> <b>одеваюсь</b></li> </ul> <p>4. Изменены параметры структуры <b>ExtraRisksInfo</b>.</p>
02.09.2015	5.3	<p>1. Функции <b>Init</b> и <b>Delnit</b> переименованы соответственно в <b>InitSpectralIM</b> и <b>DelnitSpectralIM</b>.</p> <p>2. В функциях <b>GetNextVirtFutures</b>, <b>InsertVirtFutures</b>:</p> <ul style="list-style-type: none"> <li>• удалены параметры <b>CalcDeliveryRisk</b> и <b>NClr2Delivery</b>;</li> <li>• добавлены параметры <b>ExpClearingsBF</b> и <b>ExpClearingsCC</b>.</li> </ul> <p>3. В функцию <b>RegisterFirm</b> добавлены параметры:</p> <ul style="list-style-type: none"> <li>• <b>NClr2Delivery</b> - момент начала расчета рисков экспирации;</li> <li>• <b>ExpWeight</b> - весовой коэффициент для расчета ГО;</li> <li>• <b>FirmType</b> - тип БФ;</li> <li>• <b>SettlementAccountId</b> - идентификатор РК.</li> </ul> <p>4. В функции <b>RegisterPart</b> и <b>ChangePart</b> добавлен параметр <b>ExpWeight</b>.</p> <p>5. В функции <b>SetPosition</b> и <b>ChangeOrder</b> добавлены параметры:</p> <ul style="list-style-type: none"> <li>• <b>VolatilityPartGO</b> - возвращаемое ГО клиента с учетом волатильности, пересчитанное с учетом этой позиции;</li> <li>• <b>DeliveryAndVolatilityPartGO</b> - возвращаемое ГО клиента с учетом волатильности и экспирации, пересчитанное с учетом этой позиции;</li> <li>• <b>SettlementAccountGO</b> - возвращаемое ГО РК, пересчитанное с учётом этой позиции.</li> </ul> <p>6. В структуре <b>ExtraRisksInfo</b> удален параметр <b>NULLRisk</b>.</p> <p>7. В функции <b>GetClientFutRisksArray</b>, <b>GetClientMmsRisksArray</b> и <b>GetClientMcsRisksArray</b> добавлен параметр <b>riskType</b>.</p> <p>8. Добавлена новая функция <b>RegisterSettlementAccount</b> для добавления нового РК - см. раздел 3.7.1.</p> <p>9. Изменена структура раздела <b>Архитектура библиотеки</b>. Добавлены новые подразделы:</p> <ul style="list-style-type: none"> <li>• <b>Типы конфигурации библиотеки</b> - см. раздел 2.1;</li> <li>• <b>Состав дистрибутива</b> - см. раздел 2.2;</li> <li>• <b>Принципиальная схема работы модуля SpectralIM</b> - см. раздел 2.3;</li> </ul>
15.12.2015	4.0	<p>1. Функции <b>Init</b> и <b>Delnit</b> переименованы соответственно в <b>InitSpectralIM</b> и <b>DelnitSpectralIM</b>.</p> <p>2. В функциях <b>GetNextVirtFutures</b>, <b>InsertVirtFutures</b>:</p> <ul style="list-style-type: none"> <li>• удалены параметры <b>CalcDeliveryRisk</b> и <b>NClr2Delivery</b>;</li> <li>• добавлены параметры <b>ExpClearingsBF</b> и <b>ExpClearingsCC</b>.</li> </ul> <p>3. В функцию <b>RegisterFirm</b> добавлены параметры <b>NClr2Delivery</b> и <b>ExpWeight</b>.</p> <p>4. В функции <b>RegisterPart</b> и <b>ChangePart</b> добавлен параметр <b>ExpWeight</b>.</p> <p>5. В функции <b>SetPosition</b> и <b>ChangeOrder</b> добавлены параметры <b>VolatilityPartGO</b> и <b>DeliveryAndVolatilityPartGO</b>.</p>

Дата	Версия	Изменения
		6. В структуре <b>ExtraRisksInfo</b> удален параметр <b>NULLRisk</b> .
		7. В функции <b>GetClientFutRisksArray</b> , <b>GetClientMmsRisksArray</b> и <b>GetClientMcsRisksArray</b> добавлен параметр <b>riskType</b> .

# Глава 1. Назначение библиотеки

**SpectralM** - библиотека широкого применения для расчета обеспечения на срочном рынке с C-подобным интерфейсом. Используется во внутренних сервисах ТКС Spectra и внешними клиентами в виде dll и Excel-документа.

Библиотека реализует функции загрузки с сервера биржи списка инструментов и их параметров и предоставляет API задания позиционного состояния клиентов и вычисления обеспечения по клиентам, брокерским фирмам и РК (Расчетным кодам). Загрузить можно текущие и исторические данные

# Глава 2. Архитектура библиотеки

## 2.1. Типы конфигурации библиотеки

SpectralM имеет по три конфигурации сборки для каждой из платформ (32 и 64):

- **Client** - клиентская версия библиотеки позволяет изменять параметры инструментов, модифицировать списки клиентов, а также позиции и заявки. Позволяет просматривать ГО клиентов. Клиентская версия имеет следующие ограничения:
  - требует инициализацию на сервере раздачи (проверка логина и пароля, проверка на соответствие их типу библиотеки и загрузка ограничения по количеству пользователей);
  - проверка соединения с сервером раздачи при добавлении позиций и заявок, но не чаще чем 1 раз в 20 секунд;
  - ограничение по количеству клиентов для данного логина;
  - возвращается только риск по клиентам, риск по фирме и РК всегда нулевой;
  - запрещены функции получения риск массивов (при вызове возвращают ошибку);
- **Firm** - фирменная версия библиотеки позволяет изменять параметры инструментов, просматривать ГО клиентов, фирмы и РК, получать массивы рисков. Фирменная версия имеет следующие ограничения:
  - требует инициализацию на сервере раздачи (проверка логина и пароля, проверка на соответствие их типу библиотеки и загрузка ограничения по количеству пользователей);
  - проверка соединения с сервером раздачи при добавлении позиций и заявок, но не чаще чем 1 раз в 20 секунд;
  - ограничение по количеству клиентов для данного логина;
- **Unlimited** - безлимитная версия библиотеки позволяет полный спектр действий без ограничений. Для этой версии библиотеки дополнительно поставляется внутренняя документация по риск-модулю. Безлимитная версия имеет следующие ограничения:
  - требует инициализацию на сервере раздачи (проверка логина и пароля, проверка на соответствие их типу библиотеки и загрузка ограничения по количеству пользователей);
  - ограничение по количеству клиентов для данного логина – но для этих логинов оно очень большое.

Независимо от вида библиотеки (Client/Firm/Unlimited), пользователям библиотека предоставляется всегда в виде двух DLL-файлов:

- **SpectralM.dll** - 32-разрядная версия библиотеки;
- **SpectralM64.dll** - 64-разрядная версия библиотеки.

## 2.2. Состав дистрибутива

Библиотека SpectralM предоставляется в виде дистрибутива, в который входит:

- **P2MQRouter** - роутер Plaza-II;
- **P2DB, P2DBSQLite3, P2Sys** - набор библиотек необходимых для работы роутера;
- **CLIENT\_router.ini** - настройки роутера;
- **change\_password.exe** - утилита для смены пароля;
- **SpectralM\_API.pdf** - описание API SpectralM;
- **SpectralM\_UserGuide.pdf** - руководство пользователя;
- **SpectralM.dll** - 32-х разрядная версия библиотеки;
- **SpectralM64.dll** - 64-х разрядная версия библиотеки;
- **SpectralM.ini** - ini-файл с настройками SpectralM;
- **P2Sys** - набор библиотек необходимых для работы SpectralM;
- **SpectralM.xls** - русская версия Excel - файла, демонстрирующего работу модуля SpectralM;

## 2.3. Принципиальная схема работы модуля SpectralM

1. Обмен информацией происходит с помощью платформы Plaza II. Для работы модуля должен быть настроен и запущен роутер Plaza-II.

2. Пользователь должен пройти процедуру инициализации: вызвать функцию **Init**, передав в неё путь к ini-файлу с настройками SpectralM и свой логин с паролем. В этой функции:
  - подлючается логирование;
  - создаётся соединение с роутером;
  - выполняется логин;
  - запрашивается информация о пользователе путем отправки на сервер раздачи сообщения с msg\_type=111. В Reply приходит сообщение msg\_type=112:
    - в поле part\_count - ограничение на количество пользователей;
    - в поле login\_type - тип логина CLIENT/FIRM/UNLIMITED.
3. Далее следует создать модуль расчётчика ГО. Для этого вызывается **CreateSpectralMCalculator**, который возвращает хэндлер **SpectralMCalculator**.
4. Вызов **GetInstrumentParameters** позволяет запросить инструменты с сервера раздачи на определённую дату/время (в формате "yyyy/mm/dd/hh/mm/ss" или пустая строка - текущее время).
5. Доступ к инструментам осуществляет набор функций типа **GetNextFutures**.
6. Далее регистрируются необходимые РК с помощью функции **RegisterSettlementAccount**.
7. Затем регистрируются брокерские фирмы (БФ) с помощью **RegisterFirm** (код БФ должен быть строкой из 4 символов), которые либо привязываются к одному из зарегистрированных РК, либо к dummy-РК, если в качестве идентификатора РК указать 0 или пустую строку.
8. Для каждой зарегистрированной БФ регистрируются участники с помощью **RegisterPart**. Привязка участника к БФ осуществляется по первым 4 символам его 7-значного кода.
9. Позиции и заявки выставляются при помощи функций **SetPosition** и **ChangeOrder**. При этом во всех версиях кроме **Unlimited** при вызове этих функций происходит проверка пинга с частотой раз в 20 секунд. Эти функции возвращают риски **FirmGO**, **PartGO** и **SettlementAccountGO**.
10. Функция **ClearParts** позволяет очистить зарегистрированную фирму со всеми участниками и их позициями/заявками. При этом загруженные инструменты не сбрасываются.
11. Функция **ClearAll** делает полную очистку **CCalcGOEx**: удаляет все инструменты, фирмы, участников с их позициями и заявками из **CCalcGO**.
12. Интерфейс предоставляет возможность загрузить в **SpectralMCalculator** свои инструменты при помощи следующих функций, вызывая их в следующей последовательности:
  - SetCommonParams
  - InsertCurrency
  - InsertBaseContract
  - InsertFutures
  - InsertOptionSeries
  - InsertOption
13. Для всех версий кроме клиентской SpectralM позволяет запрашивать массивы рисков:
  - GetClientVfRisksArray
  - GetClientDeliveryRisksArray
  - GetClientFutRisksArray
  - GetClientMmsRisksArray
  - GetClientMcsRisksArray
  - GetBrokerRisksArray
  - GetRiskPoints
  - GetClientFinalRisksArray
  - GetBrokerFinalRisksArray
  - GetSettlementAccountFinalRisksArray

- `GetClientFinalRisksArrayWithPositiveValue`
  - `GetBrokerFinalRisksArrayWithPositiveValue`
  - `GetSettlementAccountFinalRisksArrayWithPositiveValue`
14. По окончании работы следует удалить **SpectralMCalculator**, вызвав функцию **DestroySpectralMCalculator**.
  15. В конце сеанса работы следует вызвать функцию **DeInitSpectralM** для закрытия логов и соединений.

# Глава 3. Описание API

## 3.1. Общий перечень функций

В таблице приведен общий список функций с пометкой для какой версии библиотеки они доступны: Client (клиентская) и Firm (для БФ). Для версии специалиста (без ограничений) доступны все функции в полном объеме.

Функция	Назначение	Доступ	
		Client	Firm
Инициализация и создание модуля расчета			
InitSpectralM	Функция инициализации (см. раздел 3.3.1)	+	+
DelnitSpectralM	Функция деинициализации (см. раздел 3.3.2)	+	+
CreateSpectralMCalculator	Создает модуль расчета ГО (см. раздел 3.3.3)	+	+
DestroySpectralMCalculator	Удаляет модуль расчета ГО (см. раздел 3.3.4)	+	+
GetInstrumentParameters	Выполняет запрос списка инструментов (см. раздел 3.4.1)	+	+
Получение параметров из расчетчика			
SetOnFirstCurrency	Перемещает на первую запись в списке валют (см. раздел 3.5.1)	+	+
SetOnFirstBaseContract	Перемещает на первую запись в списке базовых контрактов (см. раздел 3.5.2)	+	+
SetOnFirstFutures	Перемещает на первую запись в списке фьючерсов (см. раздел 3.5.3)	+	+
SetOnFirstOptionSeries	Перемещает на первую запись в списке виртуальных фьючерсов (см. раздел 3.5.4)	+	+
SetOnFirstOption	Перемещает на первую запись в списке опционов (см. раздел 3.5.5)	+	+
SetOnFirstMultileg	Перемещает на первую запись в списке календарных спредов (см. раздел 3.5.6)	+	+
GetCommonParams	Возвращает общий для всех инструментов параметр - единый для всего модуля расчетчика - коэффициент учета краевых рисков. (см. раздел 3.5.7)	+	+
GetNextCurrency	Возвращает набор параметров по списку валют. (см. раздел 3.5.8)	+	+
GetNextBaseContract	Возвращает набор параметров, необходимых для расчетчика ГО, по базовым контрактам (см. раздел 3.5.9)	+	+
GetNextFutures	Возвращает набор параметров, необходимых для расчетчика ГО, по фьючерсам (см. раздел 3.5.10)	+	+
GetNextOptionSeries	Возвращает набор параметров, необходимых для расчетчика ГО, по виртуальным фьючерсам (см. раздел 3.5.11)	+	+
GetNextOption	Возвращает набор параметров, необходимых для расчетчика ГО, по опционам (см. раздел 3.5.12)	+	+
GetNextMultileg	Возвращает набор параметров, необходимых для расчетчика ГО, по календарным спредам (см. раздел 3.5.13)	+	+
Заливка данных в модуль расчетчика			
SetCommonParams	Устанавливает общий для всех инструментов параметр (см. раздел 3.6.1)	+	+
InsertCurrency	Загружает список валют в модуль расчетчика (см. раздел 3.6.2)	+	+



Функция	Назначение	Доступ	
<b>InsertBaseContract</b>	Загружает список базовых контрактов в модуль расчетчика (см. раздел 3.6.3)	+	+
<b>InsertFutures</b>	Загружает список фьючерсов в модуль расчетчика (см. раздел 3.6.4)	+	+
<b>InsertOptionSeries</b>	Загружает список опционных серий в модуль расчетчика (см. раздел 3.6.5)	+	+
<b>InsertOption</b>	Загружает список опционов в модуль расчетчика (см. раздел 3.6.6)	+	+
<b>InsertMultileg</b>	Загружает список календарных спредов в модуль расчетчика (см. раздел 3.6.7)	+	+
<b>Управление базой данных</b>			
<b>RegisterSettlementAccount</b>	Добавляет новый расчетный код (см. раздел 3.7.1)	+	+
<b>RegisterFirm</b>	Добавляет нового брокера (см. раздел 3.7.2)	+	+
<b>RegisterPart</b>	Добавляет нового клиента (см. раздел 3.7.3)	+	+
<b>ChangePart</b>	Изменяет параметры клиента (см. раздел 3.7.4)	+	+
<b>SetPosition</b>	Создание или обновление позиции по инструменту (см. раздел 3.8.1)	+	+
<b>ChangeOrder</b>	Изменение заявки по инструменту (см. раздел 3.8.2)	+	+
<b>ClearParts</b>	Очищает список клиентов (см. раздел 3.9.1)	+	+
<b>ClearAll</b>	Очищает все имеющиеся списки (см. раздел 3.9.2)	+	+
<b>Расчет рисков</b>			
<b>GetClientVfRisksArray</b>	Возвращает клиентские риски по виртуальным фьючерсам (см. раздел 3.10.1)		+
<b>GetClientDeliveryRisksArray</b>	Возвращает клиентские риски по поставочным опционам (см. раздел 3.10.2)		+
<b>GetClientFutRisksArray</b>	Возвращает клиентские риски по фьючерсам (см. раздел 3.10.3)		+
<b>GetClientMmsRisksArray</b>	Возвращает клиентские риски по межмесячным спредам (см. раздел 3.10.4)		+
<b>GetClientMcsRisksArray</b>	Возвращает клиентские риски по межконтрактным спредам (см. раздел 3.10.5)		+
<b>GetBrokerRisksArray</b>	Возвращает риски брокера (см. раздел 3.10.6)		+
<b>GetRiskPoints</b>	Возвращает диапазон ценовых точек, в которых вычисляются риски (см. раздел 3.10.7)		+
<b>GetClientFinalRisksArray</b>	Возвращает строку с итоговыми рисками для клиента. (см. раздел 3.10.8)		+
<b>GetBrokerFinalRisksArray</b>	Возвращает строку с итоговыми рисками для брокера. (см. раздел 3.10.9)		+
<b>GetSettlementAccountFinalRisksArray</b>	Возвращает строку с итоговыми рисками для РК. (см. раздел 3.10.10)		+
<b>GetClientFinalRisksArrayWithPositiveValue</b>	Возвращает строку с итоговой положительной частью риска для клиента. (см. раздел 3.10.11)		+
<b>GetBrokerFinalRisksArrayWithPositiveValue</b>	Возвращает строку с с итоговой положительной частью риска для брокера. (см. раздел 3.10.12)		+

Функция	Назначение	Доступ	
GetSettlementAccountFinalRisksArrayWithPositiveValue	Возвращает строку с с итоговой положительной частью риска для РК. (см. раздел 3.10.13)		+

## 3.2. Сервисные функции

### 3.2.1. GetSpectralIMLibraryVersion

Получает текстовое представление версии библиотеки. Формат строки не специфицируется

```
const char* GetSpectralIMLibraryVersion();
```

### 3.2.2. GetErrorCodeDescription

Получает текстовое описание ошибки по коду возврата.

```
const char* GetErrorCodeDescription(int32_t errorCode);
```

Параметр	Описание	Примечания
errorCode	Код ошибки	

## 3.3. Функции инициализации и создания модуля расчета

Перед началом работы необходимо инициализировать библиотеку, вызвав функцию **InitSpectralIM**, в которой передается путь к конфигурационному ini-файлу. Для инициализации библиотеки версий CLIENT/FIRM/UNLIMITED необходимо запустить на той же машине роутер Plaza-2 и передать функции **InitSpectralIM** логин и пароль для авторизации роутера. Параметры подключения к роутеру задаются в ini-файле. Для версии библиотеки WOAuth запуск роутера не требуется, а в качестве логина и пароля можно передать 'nullptr'.

В конце работы следует вызвать функцию **DelInitSpectralIM** для закрытия файлов диагностических журналов и соединения с роутером.

### 3.3.1. InitSpectralIM

Инициализирует модуль клиентского ГО (создает логи, устанавливает соединения, запрашивает инструменты).

При инициализации происходит подключение маршрутизатора и идентификация маршрутизатора внутри сети Plaza II, с заданным логином и паролем.

Функция не может быть сразу запущена повторно. Перед повторным запуском необходимо сначала запустить функцию **DelInitSpectralIM**.

```
int32_t InitSpectralIM(const char* IniFilePath,
                     const char* Username,
                     const char* Password);
```

Параметр	Описание	Примечания
IniFilePath	Путь к файлу настройки	
Username	Имя пользователя на сервере расчета обеспечения	
Password	Пароль пользователя на доступ к серверу	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94809** - попытка повторной инициализации
- **94814** - ошибка связи с роутером или иная ошибка при выполнении блока STARTUP
- **94815** - ошибка соединения
- **94816** - ошибка при логировании пользователя
- **94817** - несоответствие логина типу версии программы (клиент, фирма) или иное несоответствие в информации о пользователе
- **94822** - отсутствие или некорректное значение порта, указанного в ini-файле

- **94823** - отсутствие или некорректное значение таймаута, указанного в ini-файле

### 3.3.2. DeInitSpectraIM

Функция завершения работы - закрывает все логи, разрывает все соединения.

При деинициализации происходит сначала отключение внутри сети Plaza II, после отключение от самого маршрутизатора.

```
int32_t DeInitSpectraIM();
```

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94821** - вызов функции произведен до инициализации

### 3.3.3. CreateSpectraIMCalculator

Для выполнения расчетов необходимо создать калькулятор функцией **CreateSpectraIMCalculator**.

Функция возвращает число - указатель (handler) на созданный модуль расчета .

Может быть создан только один модуль расчета. При повторном вызове функции создания модуля без предварительного вызова функции **DestroySpectraIMCalculator** будет возвращаться handler уже созданного модуля.

Функция выполняется после выполнения функции **Init**.

```
int32_t CreateSpectraIMCalculator(intptr_t &SpectraIMCalculator);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94812** - не хватает памяти
- **94821** - вызов функции произведен до инициализации

### 3.3.4. DestroySpectraIMCalculator

Удаляет модуль расчета ГО. На вход подается handler этого модуля расчета.

```
int32_t DestroySpectraIMCalculator(intptr_t SpectraIMCalculator);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

## 3.4. Функции связи с сервером

Этот набор функций осуществляет доступ к инструментам, загруженным в расчётчик **CCalcGO**.

Эти функции возвращают поля через стандартные типы языка C.

### 3.4.1. GetInstrumentParameters

Выполняет запрос инструментов на указанный момент времени с сервера расчета обеспечения. Эти инструменты используются для группы функций формирования списка инструментов в форме Excel и добавляются в модуль расчетчика.

```
int32_t GetInstrumentParameters(intptr_t SpectraIMCalculator,
```

```
const char* DateTime);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>DateTime</b>	Дата и время запроса	<p>Формат: <b>YYYY/MM/DD/hh/mm/ss</b> (год/ месяц/ день/ час/ минуты/ секунды).</p> <p>Например: <b>2015/03/04/12/14/08</b>.</p> <p>Если передается нулевой указатель или пустая строка (дата и время не указаны), то запрос будет выполнен на текущий момент времени.</p>

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации
- **94995** - инструменты уже загружены, для повторной загрузки следует вызвать функцию очистки ClearAll

## 3.5. Функции для получения параметров инструментов из расчетчика

Существует набор функций, осуществляющих доступ к инструментам загруженным непосредственно в расчётчик CCalcGO. Эти функции возвращают поля через стандартные типы языка C.

Функции этой группы являются парными. Каждая пара представлена следующими функциями:

- **SetOnFirst** - установка на первом элементе;
- **GetNext** - переход к следующему элементу. Возвращает параметры определенного типа инструментов. Функции этой группы используют два параметра для передачи строк: указатель на буфер (например, FuturesCode) и его размер (например, FuturesCodeSize). Если переданного в функцию размера буфера не достаточно, возвращается ошибка 7. При этом в параметр размера помещается требуемый размер буфера. А если размера достаточно, то возвращается количество использованных байт в буфере: длина строки + 1.

### 3.5.1. SetOnFirstCurrency

Перемещает на первую запись в списке валют.

```
int32_t SetOnFirstCurrency(intptr_t SpectraIMCalculator, int32_t& CurrencyCount);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>CurrencyCount</b>	Общее количество валют	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.5.2. SetOnFirstBaseContract

Перемещает на первую запись в списке базовых контрактов.

Через переменную **BaseContractCount** возвращает общее количество базовых контрактов.

```
int32_t SetOnFirstBaseContract(intptr_t SpectraIMCalculator,
                               int32_t& BaseContractCount );
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>BaseContractCount</b>	Общее количество базовых контрактов	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.5.3. SetOnFirstFutures

Перемещает на первую запись в списке фьючерсов.

Через переменную **FuturesCount** возвращает общее количество фьючерсов.

```
int32_t SetOnFirstFutures(intptr_t SpectraIMCalculator,
                        int32_t& FuturesCount );
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>FuturesCount</b>	Общее количество фьючерсов	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.5.4. SetOnFirstOptionSeries

Перемещает на первую запись в списке виртуальных фьючерсов.

Через переменную **OptionSeriesCount** возвращает общее количество виртуальных фьючерсов.

```
int32_t SetOnFirstOptionSeries(intptr_t SpectraIMCalculator,
                              int32_t& OptionSeriesCount );
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>OptionSeriesCount</b>	Общее количество виртуальных фьючерсов	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.5.5. SetOnFirstOption

Перемещает на первую запись в списке опционов.

Через переменную **OptionCount** возвращает общее количество опционов.

```
int32_t SetOnFirstOption(intptr_t SpectraIMCalculator,
                        int32_t& OptionCount );
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>OptionCount</b>	Общее количество опционов	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.5.6. SetOnFirstMultileg

Перемещает на первую запись в списке мультилегов.

Через переменную **MultilegCount** возвращает общее количество мультилегов.

```
int32_t SetOnFirstOption(intptr_t SpectraIMCalculator,
                        int32_t& MultilegCount );
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>MultilegCount</b>	Общее количество мультилегов	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.5.7. GetCommonParams

Получает общий для всех инструментов параметр - единый для всего модуля расчетчика - коэффициент учета краевых рисков.

```
int32_t GetCommonParams( intptr_t SpectraIMCalculator,
                        double EdgeCoeff );
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>EdgeCoeff</b>	Коэффициент учета краевых рисков	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.5.8. GetNextCurrency

Возвращает набор параметров по списку валют.

```
int32_t GetNextCurrency( intptr_t SpectraIMCalculator,
                        int32_t& CurrencyId,
                        double& CurvatureRadius );
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>CurrencyId</b>	Идентификатор валюты	
<b>CurvatureRadius</b>	Радиус изменения цены валюты в процентах	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.5.9. GetNextBaseContract

Возвращает набор параметров, необходимых для расчета ГО, по базовым контрактам.

```
int32_t GetNextBaseContract( intptr_t SpectraIMCalculator,
                            char* BaseContractCode,
```

```

int32_t& BaseContractCodeSize,
char* MCSCode,
int32_t& MCSCodeSize,
int32_t& VolatNum,
int32_t& PointsNum,
Enums::MSPTYPE& MSPTYPE,
int32_t& CurrencyId,
double& ShortOptionMinimumCharge,
double& MarketRisk,
double& MarketRisk1,
double& MarketRisk2,
double& MarketRiskLow,
double& MarketRisk1Low,
double& MarketRisk2Low,
double& MarketRiskMedium,
double& MarketRisk1Medium,
double& MarketRisk2Medium,
double& MarketRiskHigh,
double& MarketRisk1High,
double& MarketRisk2High,
double& MarketRiskExceptional,
double& MarketRisk1Exceptional,
double& MarketRisk2Exceptional,
vba_int& ConcentrationLimit1,
vba_int& ConcentrationLimit2,
double& WindowSizePercentage,
Enums::OptionModel& OptionModel,
double& SpotPrice,
Enums::AssetClass& AssetClass,
double& CFRisk );

```

Параметр	Описание	Примечания
<b>SpectralMC Calculator</b>	Handler модуля расчета ГО	
<b>BaseContractCode</b>	Код базового контракта	
<b>BaseContractCodeSize</b>	Размер буфера под код базового контракта, указанный в <b>BaseContractCode</b>	
<b>MCSCode</b>	Код межконтрактного спреда	
<b>MCSCodeSize</b>	Размер буфера под код межконтрактного спреда, указанный в <b>MCSCode</b>	
<b>VolatNum</b>	Количество кривых волатильности	
<b>PointsNum</b>	Количество точек риска	
<b>MSPTYPE</b>	Тип шага цены	<b>0</b> – фиксированная стоимость шага цены <b>1</b> – стоимость шага цены, определяемая сегодняшним значением индикатора <b>2</b> – стоимость шага цены, определяемая вчерашним значением индикатора (не применяется)
<b>CurrencyId</b>	Идентификатор валюты	С его помощью рассчитывается стоимость шага цены (для MSPTYPE = 0 не заполняется).
<b>ShortOptionMinimumCharge</b>	Ставка ГО по непокрытым продажам (в рублях)	
<b>MarketRisk</b>	Ставка рыночного риска	
<b>MarketRisk1</b>	Ставка рыночного риска для первого лимита концентрации	
<b>MarketRisk2</b>	Ставка рыночного риска для второго лимита концентрации	
<b>MarketRiskLow</b>	Ставка рыночного риска для начального уровня риска клиента	
<b>MarketRisk1Low</b>	Ставка рыночного риска первого лимита концентрации для начального уровня риска клиента	
<b>MarketRisk2Low</b>	Ставка рыночного риска второго лимита концентрации для начального уровня риска клиента	

Параметр	Описание	Примечания
<b>MarketRiskMedium</b>	Ставка рыночного риска для стандартного уровня риска клиента	
<b>MarketRisk1Medium</b>	Ставка рыночного риска первого лимита концентрации для стандартного уровня риска клиента	
<b>MarketRisk2Medium</b>	Ставка рыночного риска второго лимита концентрации для стандартного уровня риска клиента	
<b>MarketRiskHigh</b>	Ставка рыночного риска для повышенного уровня риска клиента	
<b>MarketRisk1High</b>	Ставка рыночного риска первого лимита концентрации для повышенного уровня риска клиента	
<b>MarketRisk2High</b>	Ставка рыночного риска второго лимита концентрации для повышенного уровня риска клиента	
<b>MarketRiskExceptional</b>	Ставка рыночного риска для особого уровня риска клиента	
<b>MarketRisk1Exceptional</b>	Ставка рыночного риска первого лимита концентрации для особого уровня риска клиента	
<b>MarketRisk2Exceptional</b>	Ставка рыночного риска второго лимита концентрации для особого уровня риска клиента	
<b>ConcentrationLimit1</b>	Первый лимит концентрации позиции	
<b>ConcentrationLimit2</b>	Второй лимит концентрации позиции	
<b>WindowSizePercentage</b>	Диапазон сглаживания в процентах от ставки риска	Число с плавающей точкой с 6 знаками после запятой. Значение может быть в диапазоне от <b>0</b> до <b>1</b>
<b>OptionModel</b>	Опционная модель на данном БА	<ul style="list-style-type: none"> <li><b>0</b> - Black-Scholes</li> <li><b>1</b> - Bacheliers</li> </ul>
<b>SpotPrice</b>	Цена базового актива в рублях	
<b>AssetClass</b>	Тип базового актива	Перечисляемый тип. Содержит в себе классификацию типа: <ul style="list-style-type: none"> <li><b>1</b> - Акция</li> <li><b>2</b> - Валюта</li> <li><b>3</b> - Облигация</li> <li><b>4</b> - Индекс</li> <li><b>5</b> - Товар</li> <li><b>6</b> - Процентная ставка</li> <li><b>7</b> - Драг металлы</li> </ul>
<b>CFRisk</b>	Ставка риска изменения размера прогнозного денежного потока	

Возвращаемые коды ошибок (см. раздел 3.11):

- 0** - функция отработала успешно
- 94811** - ошибка входных данных
- 94821** - вызов функции произведен до инициализации

### 3.5.10. GetNextFutures

Возвращает набор параметров, необходимых для расчета ГО, по фьючерсам.

```
int32_t GetNextFutures( intptr_t SpectraIMCalculator,
```



```

Enums::ClientRiskLevel ClientRiskLevel,
char* FuturesCode,
int32_t& FuturesCodeSize,
char* BaseContractCode,
int32_t& BaseContractCodeSize,
double& Limit,
double& CalcFutPrice,
Enums::SpreadAspect& SpreadAspect,
double& MinStepPrice,
double& RealCalcPrice,
double& MinStep,
int32_t& Lot,
double& InterestRateRiskUp,
double& InterestRateRiskDown,
double& TimeToExpiration,
double& NormalizedSpot,
int32_t& Attribute,
double& MRAddOnUp,
double& MRAddOnDown,
char& EnforceIMSHalfNetting,
char* TASBaseFutCode,
int32_t& TASBaseFutCodeSize );

```

Параметр	Описание	Примечания
<b>SpectralIMCalculator</b>	Handler модуля расчета ГО	
<b>ClientRiskLevel</b>	Категория уровня риска клиента	Допустимые значения: <ul style="list-style-type: none"> <li>• <b>0</b> - Клиент с базовым уровнем риска (значение по умолчанию)</li> <li>• <b>1</b> - Клиент с Начальным Уровнем Риска (КНУР)</li> <li>• <b>2</b> - Клиент со Стандартным Уровнем Риска (КСУР)</li> <li>• <b>3</b> - Клиент с Повышенным Уровнем Риска (КПУР)</li> <li>• <b>4</b> - Клиент с Особым Уровнем Риска (КОУР)</li> </ul>
<b>FuturesCode</b>	Код фьючерса	
<b>FuturesCodeSize</b>	Размер буфера под код фьючерса, указанный в <b>FuturesCode</b>	
<b>BaseContractCode</b>	Код базового контракта	
<b>BaseContractCodeSize</b>	Размер буфера под код базового контракта, указанный в <b>BaseContractCode</b>	
<b>Limit</b>	Лимит колебания цены контракта	
<b>CalcFutPrice</b>	Расчетная цена	
<b>SpreadAspect</b>	Флаг участия в спреде	Полный, ежемесячный или никакой спред
<b>MinStepPrice</b>	Цена минимального шага (в деньгах)	
<b>RealCalcPrice</b>	Реальная расчетная цена фьючерса	
<b>MinStep</b>	Минимальный шаг изменения цены (в пунктах)	
<b>Lot</b>	Количество акций во фьючерсном контракте	
<b>InterestRateRiskUp</b>	Ставка рассогласования процентного риска в сценарии движения ставки <i>г</i> вверх.	Рассчитывается в SQL Server перед закачкой в торги методом линейной интерполяции по ключевым срокам
<b>InterestRateRiskDown</b>	Ставка рассогласования процентного риска в сценарии движения ставки <i>г</i> вниз.	Рассчитывается в SQL Server перед закачкой в торги методом линейной интерполяции по ключевым срокам
<b>TimeToExpiration</b>	Значение времени до экспирации фьючерса в долях года	
<b>NormalizedSpot</b>	Риск-параметр, означающий рассчитанную цену спота для данного базового контракта, нормализованный к размерности фьючерса	
<b>Attribute</b>	Признак спот-фьючерса	Возможные значения:

Параметр	Описание	Примечания
		<ul style="list-style-type: none"> <li>• <b>0</b> – «обычный» фьючерс</li> <li>• <b>0x80000</b> - RiskTx</li> <li>• <b>0x800000</b> - Collateral</li> <li>• <b>0x100000</b> - TAS фьючерс. (В текущей версии недоступно)</li> </ul>
<b>MRAddOnUp</b>	Надбавка Up на рыночный риск отдельного фьючерса	Надбавка на NormalizedSpot для управления гарантийным обеспечением отдельного фьючерса. Устанавливается в долях от NormalizedSpot.
<b>MRAddOnDown</b>	Надбавка Down на рыночный риск отдельного фьючерса	Надбавка на NormalizedSpot для управления гарантийным обеспечением отдельного фьючерса. Устанавливается в долях от NormalizedSpot
<b>EnforceIMSHalfNetting</b>	Признак принудительного разрыва MMC	Возможные значения: <ul style="list-style-type: none"> <li>• <b>0</b> – нет принудительного разрыва MMC</li> <li>• <b>1</b> - MMC для данного фьючерса принудительно разорван (риск считается по схеме полу-неттинг)</li> </ul>
<b>TASBaseFutCode</b>	Код опорного фьючерса	В текущей версии недоступно
<b>TASBaseFutCodeSize</b>	Размер буфера под код опорного фьючерса, указанный в <b>TASBaseFutCode</b>	В текущей версии недоступно

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.5.11. GetNextOptionSeries

Возвращает набор параметров, необходимых для расчета ГО, по опционным сериям.

```
int32_t GetNextOptionSeries( intptr_t SpectraIMCalculator,
                             char* OptionSeriesCode,
                             int32_t& OptionSeriesCodeSize,
                             char* FuturesCode,
                             int32_t& FuturesCodeSize,
                             double& SqrtT,
                             double& A_Black,
                             double& B_Black,
                             double& C_Black,
                             double& D_Black,
                             double& E_Black,
                             int8_t& UseNullVolat,
                             double& S_Black,
                             double& StrikeStep,
                             int32_t& ExpClearingsSA,
                             int32_t& ExpClearingsBF,
                             int32_t& ExpClearingsCC,
                             double& VolatilityRisk,
                             double& VolatilityRiskMismatch,
                             double& TimeToExpiration,
                             double& A_Bach,
                             double& B_Bach,
                             double& C_Bach,
                             double& D_Bach,
                             double& E_Bach,
                             double& S_Bach,
                             double& M_Bach,
                             Enums::MarginStyle& MarginStyle,
                             Enums::SettlementType& SettlementType,
                             Enums::ExerciseStyle& ExerciseStyle,
                             int8_t& SubRisk,
                             Enums::SpreadAspect& SpreadAspect,
                             int8_t& EnforceHalfNetting,
                             double& MinStep,
                             double& MinStepPrice,
```

```
int32_t& LotCoeff,
double& r,
double& InterestRateRiskUp,
double& InterestRateRiskDown,
double& FixedSpotDiscount,
double& ProjectedSpotDiscount,
double& r2,
double& InterestRate2RiskUp,
double& InterestRate2RiskDown );
```

Параметр	Описание	Примечания
<b>SpectralIMCalculator</b>	Handler модуля расчета ГО	
<b>OptionSeriesCode</b>	Код опционной серии	
<b>OptionSeriesCodeSize</b>	Размер буфера под код опционной серии, указанный в <b>OptionSeriesCode</b>	
<b>FuturesCode</b>	Код реального фьючерса	
<b>FuturesCodeSize</b>	Размер буфера под код реального фьючерса, указанный в <b>FuturesCode</b>	
<b>SqrtT</b>	Величина квадратного корня из величины времени до экспирации опционов на данный виртуальный фьючерс, выраженной в долях года.	Например: если экспирация наступает через 30 дней, то $SqrtT = \sqrt{30/365} = 0.28669$
<b>A_Black</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Блэка Шоулза
<b>B_Black</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Блэка Шоулза
<b>C_Black</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Блэка Шоулза
<b>D_Black</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Блэка Шоулза
<b>E_Black</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Блэка Шоулза
<b>UseNullVolat</b>	Флаг использования сценария нулевой волатильности	
<b>S_Black</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Блэка Шоулза
<b>StrikeStep</b>	Шаг страйка	Расстояние между двумя страйками ближайшей серии опционов
<b>ExpClearingsSA</b>	Момент начала расчета рисков экспирации для расчетного кода.	Определяет, за сколько клиринговых сессий по РК начнет блокироваться ГО, рассчитанное для всего РК по модели экспирации. Устанавливается НКЦ.
<b>ExpClearingsBF</b>	Момент начала расчета рисков экспирации для брокера.	Задается в количестве клирингов до момента экспирации.  Например: если $ExpClearingsBF=2$ , то риски начнут рассчитываться за два клиринга до момента экспирации инструмента.
<b>ExpClearingsCC</b>	Момент начала расчета рисков экспирации для клиента.	Задается в количестве клирингов до момента экспирации.  Например: если $ExpClearingsBF=2$ , то риски начнут рассчитываться за два клиринга до момента экспирации инструмента.
<b>VolatilityRisk</b>	Ставка риска волатильности	Рассчитывается в SQL Server перед закачкой в торги.  Параметр после загрузки применяется в ближайший клиринг: промежуточный или вечерний.
<b>VolatilityRiskMismatch</b>	Ставка рассогласования риска волатильности	Рассчитывается в SQL Server перед закачкой в торги.  Параметр после загрузки применяется в ближайший клиринг: промежуточный или вечерний.
<b>TimeToExpiration</b>	Значение времени до экспирации опционной серии в долях года	

Параметр	Описание	Примечания
<b>A_Bach</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Башелье
<b>B_Bach</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Башелье
<b>C_Bach</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Башелье
<b>D_Bach</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Башелье
<b>E_Bach</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Башелье
<b>S_Bach</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Башелье
<b>M_Bach</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Башелье
<b>MarginStyle</b>	Способ маржирования опциона	<ul style="list-style-type: none"> <li>0 - маржируемый</li> <li>1 – премиальный</li> </ul>
<b>SettlementType</b>	Тип опциона	<ul style="list-style-type: none"> <li>0 - расчетный</li> <li>1 - поставочный</li> </ul>
<b>ExerciseStyle</b>	Способ исполнения опциона	<ul style="list-style-type: none"> <li>0 - американский</li> <li>1 - европейский</li> </ul>
<b>SubRisk</b>	Признак учета рисков по субточкам риска фьючерса и опционов на него	
<b>SpreadAspect</b>	Флаг участия в спреде	
<b>EnforceIMSHalfNetting</b>	Признак принудительного разрыва MMC	<p>Возможные значения:</p> <ul style="list-style-type: none"> <li>0 – нет принудительного разрыва MMC</li> <li>1 - MMC для данного фьючерса принудительно разорван (риск считается по схеме полу-неттинг)</li> </ul>
<b>MinStep</b>	Размер минимального шага цены в пунктах	
<b>MinStepPrice</b>	Стоимость минимального шага цены в рублях РФ	
<b>LotCoeff</b>	Количество единиц базового актива в инструменте	
<b>r</b>	Безрисковая процентная ставка	
<b>InterestRateRiskUp</b>	Ставка рассогласования процентного риска в сценарии движения ставки r вверх.	Рассчитывается в SQL Server перед закачкой в торги методом линейной интерполяции по ключевым срокам
<b>InterestRateRiskDown</b>	Ставка рассогласования процентного риска в сценарии движения ставки r вниз.	Рассчитывается в SQL Server перед закачкой в торги методом линейной интерполяции по ключевым срокам
<b>FixedSpotDiscount</b>	Сумма дисконтированных значений выявленных денежных потоков	Размерность в единицах БА
<b>ProjectedSpotDiscount</b>	Сумма дисконтированных значений прогнозных денежных потоков	Размерность в единицах БА
<b>r2</b>	Безрисковая ставка FX2 валютной пары FX2/FX1 (для премиальных опционов на валюту); ставка дивидендной доходности q (для премиальных опционов на индекс).	
<b>InterestRate2RiskUp</b>	Ставка рассогласования процентного риска в сценарии движения ставки r2 вверх.	
<b>InterestRate2RiskDown</b>	Ставка рассогласования процентного риска в сценарии движения ставки r2 вниз.	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.5.12. GetNextOption

Возвращает набор параметров, необходимых для расчета ГО, по опционам.

```
int32_t GetNextOption( intptr_t SpectraIMCalculator,
                      char* OptionCode,
                      int32_t& OptionCodeSize,
                      char* OptionSeriesCode,
                      int32_t& OptionSeriesCodeSize,
                      double& Strike,
                      Enums::OptionType& OptionType,
                      double& CalcOptPrice );
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>OptionCode</b>	Код опциона	
<b>OptionCodeSize</b>	Размер буфера под код опциона, указанный в <b>OptionCode</b>	
<b>OptionSeriesCode</b>	Код опционной серии	
<b>OptionSeriesCodeSize</b>	Размер буфера под код опционной серии, указанный в <b>OptionSeriesCode</b>	
<b>Strike</b>	Страйк опциона	Для опционов на дальние фьючерсы, входящие в меж-месячный спрэд – страйк, преобразованный к ближнему сроку исполнения.
<b>OptType</b>	Тип опциона	<b>1</b> - PUT <b>2</b> - CALL
<b>CalcOptPrice</b>	Расчетная цена	Для маржируемых опционов

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.5.13. GetNextMultileg

Возвращает набор параметров, необходимых для расчета ГО, по мультилегам.

```
int32_t GetNextMultileg( intptr_t SpectraIMCalculator,
                        char* MultilegCode,
                        int32_t& MultilegCodeSize);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>MultilegCode</b>	Код мультилега	
<b>MultilegCodeSize</b>	Размер буфера под код мультилега, указанный в <b>MultilegCode</b>	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

## 3.6. Функции для заливки инструментов в модуль расчетчика

Функции позволяют загрузить таблицы данных в модуль расчета клиентского ГО.

### 3.6.1. SetCommonParams

Устанавливает общий для всех инструментов параметр - единый для всего модуля расчетчика - коэффициент учета краевых рисков.

```
int32_t SetCommonParams( intptr_t SpectraIMCalculator,
                        double EdgeCoeff );
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>EdgeCoeff</b>	Коэффициент учета краевых рисков	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.6.2. InsertCurrency

Загружает параметры валюты в модуль расчетчика.

```
int32_t InsertCurrency( intptr_t SpectraIMCalculator,
                      int32_t CurrencyId,
                      double CurvatureRadius);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>CurrencyId</b>	Идентификатор валюты	
<b>CurvatureRadius</b>	Радиус изменения цены валюты в процентах	Определяется в последний вечерний клиринг

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.6.3. InsertBaseContract

Добавляет базовый контракт в модуль расчетчика.

```
int32_t InsertBaseContract( intptr_t SpectraIMCalculator,
                          const char* BaseContractCode,
                          const char* MCSCode,
                          int32_t VolatNum,
                          Enums::MSPTType MSPTType,
                          int32_t CurrencyId6,
                          double ShortOptionMinimumCharge,
                          double MarketRisk,
                          double MarketRisk1,
                          double MarketRisk2,
                          double MarketRiskLow,
                          double MarketRisk1Low,
                          double MarketRisk2Low,
                          double MarketRiskMedium,
                          double MarketRisk1Medium,
                          double MarketRisk2Medium,
```

```
double MarketRiskHigh,
double MarketRisk1High,
double MarketRisk2High,
double MarketRiskExceptional,
double MarketRisk1Exceptional,
double MarketRisk2Exceptional,
vba_int ConcentrationLimit1,
vba_int ConcentrationLimit2,
int32_t RiskPointsN,
double WindowSizePercentage,
Enums::OptionModel OptionModel,
double SpotPrice,
Enums::AssetClass AssetClass,
double CFRisk );
```

Параметр	Описание	Примечания
<b>SpectralMCalculator</b>	Handler модуля расчета ГО	
<b>Code</b>	Код базового контракта	
<b>MCSCode</b>	Код межконтрактного спреда	
<b>VolatNum</b>	Количество кривых волатильности	
<b>MSPTYPE</b>	Тип стоимости шага цены: <ul style="list-style-type: none"> <li>• 0 - фиксированный,</li> <li>• 1 - переменный</li> </ul>	
<b>CurrencyId</b>	Идентификатор валюты	
<b>ShortOptionMinimumCharge</b>	Ставка ГО по непокрытым продажам в долях от ставки риска первого уровня	Минимальное требование за «непокрытую» продажу. Устанавливается в долях от ставки рыночного риска MarketRisk.
<b>MarketRisk</b>	Ставка рыночного риска	
<b>MarketRisk1</b>	Ставка рыночного риска для первого лимита концентрации	
<b>MarketRisk2</b>	Ставка рыночного риска для второго лимита концентрации	
<b>MarketRiskLow</b>	Ставка рыночного риска для начального уровня риска клиента	
<b>MarketRisk1Low</b>	Ставка рыночного риска первого лимита концентрации для начального уровня риска клиента	
<b>MarketRisk2Low</b>	Ставка рыночного риска второго лимита концентрации для начального уровня риска клиента	
<b>MarketRiskMedium</b>	Ставка рыночного риска для стандартного уровня риска клиента	
<b>MarketRisk1Medium</b>	Ставка рыночного риска первого лимита концентрации для стандартного уровня риска клиента	
<b>MarketRisk2Medium</b>	Ставка рыночного риска второго лимита концентрации для стандартного уровня риска клиента	
<b>MarketRiskHigh</b>	Ставка рыночного риска для повышенного уровня риска клиента	
<b>MarketRisk1High</b>	Ставка рыночного риска первого лимита концентрации для повышенного уровня риска клиента	
<b>MarketRisk2High</b>	Ставка рыночного риска второго лимита концентрации для повышенного уровня риска клиента	
<b>MarketRiskExceptional</b>	Ставка рыночного риска для особого уровня риска клиента	
<b>MarketRisk1Exceptional</b>	Ставка рыночного риска первого лимита концентрации для особого уровня риска клиента	

Параметр	Описание	Примечания
<b>MarketRisk2Exceptional</b>	Ставка рыночного риска второго лимита концентрации для особого уровня риска клиента	
<b>ConcentrationLimit1</b>	Первый лимит концентрации позиции	
<b>ConcentrationLimit2</b>	Второй лимит концентрации позиции	
<b>RiskPointsN</b>	Количество точек расчета риска	Расчет производится по формуле $2*N+1$
<b>WindowSizePercentage</b>	Коэффициент для определения размера окна сглаживания при маржировании межконтрактного спреда	Число с плавающей точкой с 6 знаками после запятой. Значение может быть в диапазоне от <b>0</b> до <b>1</b>
<b>OptionModel</b>	Опционная модель на данном БА	<ul style="list-style-type: none"> <li><b>0</b> - Black-Scholes</li> <li><b>1</b> - Bacheliers</li> </ul>
<b>SpotPrice</b>	Расчетная цена контракта в рублях	
<b>AssetClass</b>	Тип базового актива	Перечисляемый тип. Содержит в себе классификацию типа: <ul style="list-style-type: none"> <li><b>1</b> - Акция</li> <li><b>2</b> - Валюта</li> <li><b>3</b> - Облигация</li> <li><b>4</b> - Индекс</li> <li><b>5</b> - Товар</li> <li><b>6</b> - Процентная ставка</li> <li><b>7</b> - Драг металлы</li> </ul>
<b>CFRisk</b>	Ставка риска изменения размера прогнозного денежного потока	

Возвращаемые коды ошибок (см. раздел 3.11):

- 0** - функция отработала успешно
- 94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- 94811** - ошибка входных данных
- 94821** - вызов функции произведен до инициализации

### 3.6.4. InsertFutures

Добавляет фьючерсный контракт в модуль расчетчика.

```
int32_t InsertFutures( intptr_t SpectraIMCalculator,
                      const char* FuturesCode,
                      const char* BaseContractCode,
                      double CalcFutPrice,
                      Enums::SpreadAspect SpreadAspect,
                      double MinStepPrice,
                      double RealCalcPrice,
                      double MinStep,
                      int32_t Lot,
                      double InterestRateRiskUp,
                      double InterestRateRiskDown,
                      double TimeToExpiration,
                      double NormalizedSpot,
                      int32_t Attribute,
                      double MRAddOnUp,
                      double MRAddOnDown,
                      char EnforceIMSHalfNetting,
                      const char* TASBaseFutCode );
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>FuturesCode</b>	Код фьючерсного контракта	



Параметр	Описание	Примечания
<b>BaseContractCode</b>	Код базового контракта	
<b>CalcFutPrice</b>	Расчетная цена	
<b>SpreadAspect</b>	Признак вхождения в межмесячный спред	
<b>MinStepPrice</b>	Стоимость минимального шага цены в рублях РФ	
<b>RealCalcPrice</b>	Реальная расчетная цена фьючерса	
<b>MinStep</b>	Размер минимального шага цены в пунктах	
<b>Lot</b>	Количество акций во фьючерсном контракте	
<b>InterestRateRiskUp</b>	Ставка рассогласования процентного риска в сценарии движения ставки $r$ вверх.	Рассчитывается в SQL Server перед закачкой в торги методом линейной интерполяции по ключевым срокам
<b>InterestRateRiskDown</b>	Ставка рассогласования процентного риска в сценарии движения ставки $r$ вниз.	Рассчитывается в SQL Server перед закачкой в торги методом линейной интерполяции по ключевым срокам
<b>TimeToExpiration</b>	Значение времени до экспирации фьючерса в долях года	
<b>NormalizedSpot</b>	Риск-параметр, означающий рассчитанную цену спота для данного базового контракта, нормализованный к размерности фьючерса	
<b>Attribute</b>	Признак спот-фьючерса	Возможные значения: <ul style="list-style-type: none"> <li>• <b>0</b> – «обычный» фьючерс</li> <li>• <b>0x80000</b> - RiskTx</li> <li>• <b>0x800000</b> - Collateral</li> <li>• <b>0x1000000</b> - TAS фьючерс. (В текущей версии недоступно)</li> </ul>
<b>MRAddOnUp</b>	Надбавка Up на рыночный риск отдельного фьючерса	Надбавка на NormalizedSpot для управления гарантийным обеспечением отдельного фьючерса. Устанавливается в долях от NormalizedSpot.
<b>MRAddOnDown</b>	Надбавка Down на рыночный риск отдельного фьючерса	Надбавка на NormalizedSpot для управления гарантийным обеспечением отдельного фьючерса. Устанавливается в долях от NormalizedSpot
<b>EnforceIMSHalfNetting</b>	Признак принудительного разрыва MMC	Возможные значения: <ul style="list-style-type: none"> <li>• <b>0</b> – нет принудительного разрыва MMC</li> <li>• <b>1</b> - MMC для данного фьючерса принудительно разорван (риск считается по схеме полу-неттинг)</li> </ul>
<b>TASBaseFutCode</b>	Код опорного фьючерса	В текущей версии недоступно

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.6.5. InsertOptionSeries

Добавляет опционную серию в модуль расчетчика.

```
int32_t InsertOptionSeries( intptr_t SpectraIMCalculator,
                           const char* OptionSeriesCode,
                           const char* FuturesCode,
                           double A_Black,
                           double B_Black,
```

```

double C_Black,
double D_Black,
double E_Black,
int8_t UseNullVolat,
double S_Black,
double StrikeStep,
int32_t ExpClearingsSA,
int32_t ExpClearingsBF,
int32_t ExpClearingsCC,
double VolatilityRisk,
double VolatilityRiskMismatch,
double TimeToExpiration,
double A_Bach,
double B_Bach,
double C_Bach,
double D_Bach,
double E_Bach,
double S_Bach,
double M_Bach,
Enums::MarginStyle MarginStyle,
Enums::SettlementType SettlementType,
Enums::ExerciseStyle ExerciseStyle,
int8_t SubRisk,
Enums::SpreadAspect SpreadAspect,
int8_t EnforceHalfNetting,
double MinStep,
double MinStepPrice,
int32_t LotCoeff,
double r,
double InterestRateRiskUp,
double InterestRateRiskDown,
double FixedSpotDiscount,
double ProjectedSpotDiscount,
double r2,
double InterestRate2RiskUp,
double InterestRate2RiskDown );

```

Параметр	Описание	Примечания
<b>SpectralIMCalculator</b>	Handler модуля расчета ГО	
<b>OptionSeriesCode</b>	Код опционной серии	
<b>FuturesCode</b>	Код реального фьючерса	
<b>A_Black</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Блэка Шоулза
<b>B_Black</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Блэка Шоулза
<b>C_Black</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Блэка Шоулза
<b>D_Black</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Блэка Шоулза
<b>E_Black</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Блэка Шоулза
<b>UseNullVolat</b>	Флаг использования сценария нулевой волатильности	
<b>S_Black</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Блэка Шоулза
<b>StrikeStep</b>	Шаг страйка	Расстояние между двумя страйками ближайшей серии опционов внутри диапазона [-L;+L]
<b>ExpClearingsSA</b>	Момент начала расчета рисков экспирации для расчетного кода.	Определяет, за сколько клиринговых сессий по РК начнет блокироваться ГО, рассчитанное для всего РК по модели экспирации. Устанавливается НКЦ.
<b>ExpClearingsBF</b>	Момент начала расчета рисков экспирации для брокера.	<p>Задается в количестве клирингов до момента экспирации.</p> <p>Например: если ExpClearingsBF=2, то риски начнут рассчитываться за два клиринга до момента экспирации инструмента.</p>

Параметр	Описание	Примечания
<b>ExpClearingsCC</b>	Момент начала расчета рисков экспирации для клиента.	Задается в количестве клирингов до момента экспирации.  Например: если ExpClearingsCC=2, то риски начнут рассчитываться за два клиринга до момента экспирации инструмента.
<b>VolatilityRisk</b>	Ставка риска волатильности	Рассчитывается в SQL Server перед закачкой в торги.  Параметр после загрузки применяется в ближайший клиринг: промежуточный или вечерний.
<b>VolatilityRiskMismatch</b>	Ставка рассогласования риска волатильности	Рассчитывается в SQL Server перед закачкой в торги.  Параметр после загрузки применяется в ближайший клиринг: промежуточный или вечерний.
<b>TimeToExpiration</b>	Значение времени до экспирации опционной серии	
<b>A_Bach</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Башелье
<b>B_Bach</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Башелье
<b>C_Bach</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Башелье
<b>D_Bach</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Башелье
<b>E_Bach</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Башелье
<b>S_Bach</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Башелье
<b>M_Bach</b>	Параметр кривой волатильности	Массив параметров функции расчета волатильности по формуле Башелье
<b>MarginStyle</b>	Способ маржирования опциона	<ul style="list-style-type: none"> <li>0 - маржируемый</li> <li>1 – премиальный</li> </ul>
<b>SettlementType</b>	Тип опциона	<ul style="list-style-type: none"> <li>0 - расчетный</li> <li>1 - поставочный</li> </ul>
<b>ExerciseStyle</b>	Способ исполнения опциона	<ul style="list-style-type: none"> <li>0 - американский</li> <li>1 - европейский</li> </ul>
<b>SubRisk</b>	Признак учета рисков по субточкам риска фьючерса и опционов на него	
<b>SpreadAspect</b>	Признак вхождения в межмесячный спред	
<b>EnforceIMSHalfNetting</b>	Признак принудительного разрыва MMC	<p>Возможные значения:</p> <ul style="list-style-type: none"> <li>0 – нет принудительного разрыва MMC</li> <li>1 - MMC для данного фьючерса принудительно разорван (риск считается по схеме полу-неттинг)</li> </ul>
<b>MinStep</b>	Размер минимального шага цены в пунктах	
<b>MinStepPrice</b>	Стоимость минимального шага цены в рублях РФ	
<b>LotCoeff</b>	Количество единиц базового актива в инструменте	
<b>r</b>	Безрисковая процентная ставка	
<b>InterestRateRiskUp</b>	Ставка рассогласования процентного риска в сценарии движения ставки $r$ вверх.	Рассчитывается в SQL Server перед закачкой в торги методом линейной интерполяции по ключевым срокам

Параметр	Описание	Примечания
<b>InterestRateRiskDown</b>	Ставка рассогласования процентного риска в сценарии движения ставки $r$ вниз.	Рассчитывается в SQL Server перед закачкой в торги методом линейной интерполяции по ключевым срокам
<b>FixedSpotDiscount</b>	Сумма дисконтированных значений выявленных денежных потоков	Размерность в единицах БА
<b>ProjectedSpotDiscount</b>	Сумма дисконтированных значений прогнозных денежных потоков	Размерность в единицах БА
<b>r2</b>	Безрисковая ставка FX2 валютной пары FX2/FX1 (для премиальных опционов на валюту); ставка дивидендной доходности $q$ (для премиальных опционов на индекс).	
<b>InterestRate2RiskUp</b>	Ставка рассогласования процентного риска в сценарии движения ставки $r2$ вверх.	
<b>InterestRate2RiskDown</b>	Ставка рассогласования процентного риска в сценарии движения ставки $r2$ вниз.	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.6.6. InsertOption

Добавляет опционный инструмент в модуль расчетчика.

При добавлении опциона автоматически должны предрасчитываться теоретические цены и экстрариски.

```
int32_t InsertOption( intptr_t SpectraIMCalculator,
                    const char* OptionCode,
                    const char* OptionSeriesCode,
                    double Strike,
                    Enums::OptionType OptionType,
                    double CalcOptPrice );
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>OptionCode</b>	Код опциона	
<b>OptionSeriesCode</b>	Код опционной серии	
<b>Strike</b>	Страйк опциона.	Для опционов на дальние фьючерсы, входящие в меж-месячный спред – страйк, преобразованный к ближнему сроку исполнения.
<b>OptType</b>	Тип опциона	<b>1</b> - PUT <b>2</b> - CALL
<b>CalcOptPrice</b>	Расчетная цена	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.6.7. InsertMultileg

Добавляет календарный спред (мультилеги) в модуль расчетчика.

```
int32_t InsertMultileg( intptr_t SpectraIMCalculator,
                      const char* MultilegCode );
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>MultilegCode</b>	Код календарного спреда	<p>Формат: &lt;код базового контракта&gt;-&lt;месяц.год&gt;-&lt;месяц.год&gt;</p> <p>Например: <b>RTS-6.19-9.19</b></p> <p>где: <b>RTS-6.19</b> - код ближней ноги (имя-дата)</p> <p><b>RTS-9.19</b> - код дальней ноги (имя-дата)</p>

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации
- **94792** - мультилег уже существует

### 3.6.8. InsertBrokerBaseContractParam

Добавляет коэффициент ГО в разрезе брокера и базового актива в модуль расчетчика.

```
int32_t InsertBrokerBaseContractParam(intptr_t SpectraIMCalculator,
                                     const char* Code,
                                     const char* CodeBC,
                                     double GORatio);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>Code</b>	Код Брокерской фирмы	
<b>CodeBC</b>	Код базового контракта	
<b>GORatio</b>	Коэффициент ГО по базовому контракту для всех клиентов данной БФ	Значение коэффициента от 0 до 10

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

## 3.7. Функции для управления данными о фирмах и клиентах

### 3.7.1. RegisterSettlementAccount

Добавляет новый расчетный код (РК).

```
int32_t RegisterSettlementAccount( intptr_t SpectraIMCalculator,
                                   const char* SettlementAccountId,
                                   Enums::MarginRule MarginRule,
                                   Enums::SpreadMarginType CalendarSpreadMarginType,
                                   Enums::SpreadMarginType InterContractSpreadMarginType,
                                   double ShortOptionMinimumChargeRatio);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	Указатель на объект SpectraIM Calculator (CCalcGOEx)

Параметр	Описание	Примечания
<b>SettlementAccountId</b>	Идентификатор регистрируемого РК	Строка длиной до 12 символов
<b>MarginRule</b>	Правило маржирования для регистрируемого РК	Допустимые значения: <ul style="list-style-type: none"> <li>• <b>2</b> – гросс</li> <li>• <b>3</b> – полунетто</li> <li>• <b>4</b> – нетто</li> </ul>
<b>CalendarSpreadMarginType</b>	Тип маржирования календарного спреда	Допустимые значения: <ul style="list-style-type: none"> <li>• <b>3</b> – полунеттинг</li> <li>• <b>4</b> – неттинг (по умолчанию)</li> </ul>
<b>InterContractSpreadMarginType</b>	Правило маржирования для контрактов, входящих в межконтрактный спред	Допустимые значения: <ul style="list-style-type: none"> <li>• <b>3</b> – полунеттинг</li> <li>• <b>4</b> – неттинг (по умолчанию)</li> </ul>
<b>ShortOptionMinimumChargeRatio</b>	Коэффициент учета сценария SOMC	Для РК всегда <b>1</b>

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.7.2. RegisterFirm

Добавляет новую фирму-брокера.

Брокерские фирмы (БФ) либо привязываются к одному из зарегистрированных РК, либо к dummy-РК, если в качестве идентификатора РК указан 0 или пустая строка.

```
int32_t RegisterFirm( intptr_t SpectraIMCalculator,
                    const char* BrokerCode,
                    int32_t NClr2Delivery,
                    double ExpWeight,
                    double ExpWeightClientDefault,
                    Enums::BrokerAccountType BrokerAccountType,
                    const char* SettlementAccountId,
                    Enums::SpreadMarginType CalendarSpreadMarginType,
                    Enums::MarginRule MarginRule,
                    Enums::SpreadMarginType InterContractSpreadMarginType,
                    double ShortOptionMinimumChargeRatio);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>BrokerCode</b>	Код брокерской фирмы	Код должен содержать 4 символа.
<b>NClr2Delivery</b>	Момент начала расчета рисков экспирации	Задается в количестве клирингов до момента экспирации.  Задается для брокера Биржей и не может быть изменено самим брокером.
<b>ExpWeight</b>	Весовой коэффициент для расчета ГО с учетом волатильности и экспирации	Коэффициент, с которым должны учитываться риски автоэкспирации
<b>ExpWeightClientDefault</b>	Весовой коэффициент для расчета ГО с включенным сценарием поставки в итоговом ГО для клиентов БФ по умолчанию	Весовой коэффициент для клиентов брокера, используемый по умолчанию.
<b>BrokerAccountType</b>	Тип брокерской фирмы	Допустимые значения: <ul style="list-style-type: none"> <li>• <b>0</b> – собственная</li> <li>• <b>1</b> – клиентская</li> <li>• <b>2</b> – ДУ</li> </ul>

Параметр	Описание	Примечания
		• <b>3</b> - ОБФ
<b>SettlementAccountId</b>	Идентификатор Расчетного кода, которому принадлежит регистрируемая БФ	РК с указанным идентификатором должен быть зарегистрирован прежде его использования в функции <b>RegisterFirm</b>
<b>CalendarSpreadMarginType</b>	Тип календарного спреда	Допустимые значения: • <b>3</b> – полунеттинг • <b>4</b> – неттинг
<b>marginRule</b>	Правило маржирования для регистрируемого РК	Допустимые значения: • <b>2</b> – гросс • <b>3</b> – полунетто • <b>4</b> – нетто
<b>InterContractSpreadMarginType</b>	Правило маржирования для контрактов, входящих в межконтрактный спред	Допустимые значения: • <b>3</b> – полунеттинг • <b>4</b> – неттинг (по умолчанию)
<b>ShortOptionMinimumChargeRatio</b>	Коэффициент учета сценария SOMC	Для РК всегда <b>1</b>

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.7.3. RegisterPart

Добавляет клиентов брокера, который определяется функцией **RegisterFirm**

Привязка участника к БФ осуществляется по первым 4 символам его 7-значного кода.

```
int32_t RegisterPart( intptr_t SpectraIMCalculator,
                    const char* ClientCode,
                    int32_t NoFutDiscount,
                    int32_t NClr2Delivery,
                    double ExpWeight,
                    Enums::SpreadMarginType CalendarSpreadMarginType,
                    Enums::SpreadMarginType InterContractSpreadMarginType,
                    double GORatio,
                    double ShortOptionMinimumChargeRatio,
                    Enums::ClientRiskLevel ClientRiskLevel );
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>ClientCode</b>	Код клиента	Должен содержать 7 символов, где первые 4 символа код брокера, а остальные 3 - код клиента.
<b>NoFutDiscount</b>	Признак учета скидки по фьючерсу для данного клиента	<b>1</b> - скидка учитывается при расчете <b>0</b> - скидка не учитывается при расчете
<b>NClr2Delivery</b>	Момент начала расчетов рисков экспирации - количество клирингов, за которое включаются сценарии автоэкспирации	При заданном (не ноль) <b>NClr2Delivery</b> у виртуального фьючерса, брокер может включить расчёт рисков поставки до того момента, как начнется расчет этих рисков по клиенту.  Если значение этого параметра установлено <b>"-1"</b> , то берется значение по умолчанию для всех клиентов - параметр <b>NClr2Delivery</b> из таблицы <b>RegisterFirm</b> .

Параметр	Описание	Примечания
<b>ExpWeight</b>	Коэффициент для расчета ГО, с которым должны учитываться риски автоэкспирации	Устанавливается брокером для своих клиентов.  Если значение этого параметра установлено "-1", то берется значение по умолчанию для всех клиентов - параметр <b>ExpWeight</b> из таблицы <b>RegisterFirm</b> .
<b>CalendarSpreadMarginType</b>	Тип календарного спреда	Допустимые значения: • <b>3</b> – полунеттинг • <b>4</b> – неттинг
<b>InterContractSpreadMarginType</b>	Правило маржирования для контрактов, входящих в межконтрактный спред	Допустимые значения: • <b>3</b> – полунеттинг • <b>4</b> – неттинг (по умолчанию)
<b>GORatio</b>	Коэффициент клиентского ГО	
<b>ShortOptionMinimumChargeRatio</b>	Коэффициент учета сценария SOMC	Для РК всегда <b>1</b>
<b>ClientRiskLevel</b>	Категория уровня риска клиента	Допустимые значения: • <b>0</b> - Клиент с базовым уровнем риска (значение по умолчанию) • <b>1</b> - Клиент с Начальным Уровнем Риска (КНУР) • <b>2</b> - Клиент со Стандартным Уровнем Риска (КСУР) • <b>3</b> - Клиент с Повышенным Уровнем Риска (КПУР) • <b>4</b> - Клиент с Особым Уровнем Риска (КОУР)

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94818** - превышение числа клиентов
- **94821** - вызов функции произведен до инициализации

### 3.7.4. ChangePart

Функция изменения параметров клиента.

```
int32_t ChangePart( intptr_t SpectraIMCalculator,
                  const char* ClientCode,
                  int32_t NoFutDiscount,
                  int32_t NClr2Delivery,
                  double Expweight);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>ClientCode</b>	Код клиента	Должен содержать 7 символов, где первые 4 символа код брокера, а остальные 3 - код клиента.
<b>NoFutDiscount</b>	Признак учета скидки по фьючерсу для данного клиента	<b>1</b> - скидка учитывается при расчете <b>0</b> - скидка не учитывается при расчете
<b>NClr2Delivery</b>	Кол-во клирингов, за которое включаются сценарии автоэкспирации	При взведённом (не ноль) <b>NClr2Delivery</b> у виртуального фьючерса, брокер может включить расчёт рисков поставки до того момента, как начнется расчет этих рисков по клиенту



Параметр	Описание	Примечания
<b>ExpWeight</b>	Коэффициент для расчета ГО, с которым должны учитываться риски автоэкспирации	Устанавливается брокером для своих клиентов. Если значение этого параметра установлено "-1", то берется значение по умолчанию для всех клиентов - параметр <b>ExpWeight</b> из таблицы <b>RegisterFirm</b> .

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

## 3.8. Функции для управления позициями и заявками

### 3.8.1. SetPosition

Создание или обновление позиции по инструменту.

```
int32_t SetPosition( intptr_t SpectraIMCalculator,
                    const char *Code,
                    const char *InstrCode,
                    vba_int Quantity,
                    double WAPrice,
                    double* VolatilityPartGO,
                    double* DeliveryAndVolatilityPartGO,
                    double* PartGO,
                    double* FirmGO,
                    double* SettlementAccountGO,
                    Enums::AccountHierarchyRecordType accountHierarchy,
                    vba_int AfterLastClearingQuantity);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>Code</b>	Код клиента, БФ или РК	Строка должна содержать семизначный код клиента
<b>InstrCode</b>	Код инструмента, заявка по которому должна быть изменена	
<b>Quantity</b>	Количество инструмента - объем позиции	Количество купленных контрактов минус количество проданных контрактов
<b>WAPrice</b>	(Weighted average price) средневзвешенная цена позиции (пункты)	Используется для фьючерсной позиции. Если этот параметр равен нулю, то будет использоваться расчетная цена инструмента.
<b>VolatilityPartGO</b>	Возвращаемое ГО клиента с учетом волатильности, пересчитанное с учетом этой позиции	ГО по клиенту, после изменения/добавления заявки, БЕЗ учёта рисков автоэкспирации
<b>DeliveryAndVolatilityPartGO</b>	Возвращаемое ГО клиента с учетом волатильности и экспирации, пересчитанное с учетом этой позиции	ГО по клиенту, после изменения/добавления заявки, С учётом рисков автоэкспирации
<b>PartGO</b>	Возвращаемое ГО клиента, пересчитанное с учетом этой позиции. Это суммарное ГО, состоящее из ГО с учетом волатильности и из ГО с учетом волатильности и экспирации. При расчете учитываются весовые коэффициенты этих ГО.	
<b>FirmGO</b>	Возвращаемое ГО брокера, пересчитанное с учетом этой позиции	Для клиентской версии возвращается <b>0</b>
<b>SettlementAccountGO</b>	Возвращаемое ГО РК, пересчитанное с учетом этой позиции	Для клиентской версии возвращается <b>0</b>
<b>AccountHierarchyRecordType</b>	Тип клиента	Допустимые значения: • <b>0</b> – РК

Параметр	Описание	Примечания
		<ul style="list-style-type: none"> <li><b>1</b> – брокер</li> <li><b>2</b> – клиент</li> </ul>
<b>AfterLastClearingQuantity</b>	Объём позиции на момент последнего клиринга (ПК или ВК)	

Возвращаемые коды ошибок (см. раздел 3.11):

- 0** - функция отработала успешно
- 94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- 94811** - ошибка входных данных
- 94820** - нет соединения. При возникновении этой ошибки для дальнейшей работы пользователь должен вызвать сначала функцию **Delnit**, а затем **Init**.
- 94821** - вызов функции произведен до инициализации

### 3.8.2. ChangeOrder

Создание или изменение заявки по инструменту.

```
int32_t ChangeOrder( intptr_t SpectraIMCalculator,
                    const char* Code,
                    const char *InstrCode,
                    int32_t IsBuy,
                    double Price,
                    vba_int DeltaQuantity,
                    double* VolatilityPartGO,
                    double* DeliveryAndVolatilityPartGO,
                    double* PartGO,
                    double* FirmGO,
                    double* SettlementAccountGO);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>Code</b>	Код клиента	Строка, которая должна содержать семизначный код клиента
<b>InstrCode</b>	Код инструмента	
<b>IsBuy</b>	Направление заявки	<b>не 0</b> - покупка <b>0</b> - продажа
<b>Price</b>	Цена за единицу инструмента	Если этот параметр равен нулю, то будет использоваться расчетная цена инструмента.
<b>DeltaQuantity</b>	Количество инструмента, на которое должен быть изменен объем заявки	
<b>VolatilityPartGO</b>	Возвращаемое ГО клиента с учетом волатильности, пересчитанное с учетом этой позиции	ГО по клиенту, после изменения/добавления заявки, БЕЗ учёта рисков автоэкспирации
<b>DeliveryAndVolatilityPartGO</b>	Возвращаемое ГО клиента с учетом волатильности и экспирации, пересчитанное с учетом этой позиции	ГО по клиенту, после изменения/добавления заявки, С учётом рисков автоэкспирации
<b>PartGO</b>	Возвращаемое ГО клиента, состоящее из ГО с учетом волатильности и из ГО с учетом волатильности и экспирации. При расчете учитываются весовые коэффициенты этих ГО.	
<b>FirmGO</b>	Возвращаемое ГО брокера, пересчитанное с учетом изменений параметров заявки	Для клиентской версии возвращается <b>0</b>
<b>SettlementAccountGO</b>	Возвращаемое ГО РК, пересчитанное с учетом изменений параметров заявки	Для клиентской версии возвращается <b>0</b>

Возвращаемые коды ошибок (см. раздел 3.11):

- 0** - функция отработала успешно

- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94820** - нет соединения. При возникновении этой ошибки для дальнейшей работы пользователь должен вызвать сначала функцию **Delnit**, а затем **Init**.
- **94821** - вызов функции произведен до инициализации

## 3.9. Функции очистки

### 3.9.1. ClearParts

Очищает весь список клиентов.

```
int32_t ClearParts(intptr_t SpectraIMCalculator);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

### 3.9.2. ClearAll

Очищает все данные в расчетчике (списки клиентов, инструментов и пр.).

```
int32_t ClearAll(intptr_t SpectraIMCalculator);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94821** - вызов функции произведен до инициализации

## 3.10. Функции для получения массивов рисков клиентов

Для всех версий кроме клиентской **SpectraIM** позволяет запрашивать массивы рисков с помощью функций, описанных в этом разделе.

Функции используются для заполнения таблицы на вкладке **Массив рисков** в форме Excel для клиентского ГО.

Первые пять функций используются для построения графиков на вкладке **Массив рисков** в форме Excel для клиентского ГО.

### 3.10.1. GetClientVfRisksArray

Возвращает клиентские риски по виртуальным фьючерсам. Если у клиента нет опционов, то функция ничего не возвращает.

Функция через аргумент **arr** возвращает верхнюю часть колонки в таблице на вкладке **Массив рисков**, а через аргумент **xInfo** - нижнюю часть колонки в этой таблице.

Функция вызывается два раза:

1. Вызов с **pointsCount <= 0** и **scenarioCount <= 0** – возвращает в эти две переменные значения: «число точек расчёта риска» и «число сценариев риска».
2. Вызов с заданным **pointsCount > 0** и **scenarioCount > 0 (1, 2, ...)** – возвращает в переменную **arr** массив из **pointsCount** - точек расчёта риска всех сценариев (количество сценариев равно **scenarioCount**). Массив должен быть предварительно зарезервирован.

рован в размере, соответствующем количеству точек расчета риска **pointsCount**, возвращённому из предыдущего вызова. Значение **pointsCount** из второго вызова не должно превышать значение из первого вызова.

```
int32_t GetClientVfRisksArray(intptr_t SpectraIMCalculator,
                             eChooseRiskType choosenRiskType,
                             const char* instrCode,
                             const char* Code,
                             double* arr,
                             int32_t& pointsCount,
                             int32_t& scenarioCount,
                             ExtraRisksInfo* xInfo);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>choosenRiskType</b>	Тип возвращаемого риска: <ul style="list-style-type: none"> <li>• <b>0</b> - риск волатильности</li> <li>• <b>1</b> - риск волатильности и экспирации</li> </ul>	
<b>instrCode</b>	Код брокерской фирмы	
<b>Code</b>	Код клиента	Семизначный код клиента
<b>arr</b>	Зарезервированный массив	
<b>pointsCount</b>	Количество точек расчета риска	
<b>scenarioCount</b>	Количество сценариев	Число сценариев для виртуального фьючерса задается в спецификации базового контракта.  Число сценариев определяет количество столбиков в таблице на вкладке <b>Массив рисков</b> .
<b>xInfo</b>	Дополнительная информация о массиве рисков, хранящаяся в структуре <b>ExtraRisksInfo</b> (см. раздел 3.10.14)	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94819** - функция недоступна (для клиентской версии)
- **94821** - вызов функции произведен до инициализации

### 3.10.2. GetClientDeliveryRisksArray

Возвращает клиентские риски по поставочным опционам (дополнительные риски). Если у клиента нет поставочных опционов, то функция ничего не возвращает.

Функция через аргумент **arr** возвращает верхнюю часть колонки в таблице на вкладке **Массив рисков**, а через аргумент **xInfo** - нижнюю часть колонки в этой таблице.

Функция вызывается два раза:

1. Вызов с **pointsCount <= 0** и **scenarioCount <= 0** – возвращает в эти две переменные значения: «число точек расчёта риска» и «число сценариев риска».
2. Вызов с заданным **pointsCount > 0** и **scenarioCount > 0 (1, 2, ...)** – возвращает в переменную **arr** массив из **pointsCount** - точек расчёта риска всех сценариев (количество сценариев равно **scenarioCount**). Массив должен быть предварительно зарезервирован в размере, соответствующем количеству точек расчета риска **pointsCount**, возвращённому из предыдущего вызова. Значение **pointsCount** из второго вызова не должно превышать значение из первого вызова.

```
int32_t GetClientDeliveryRisksArray(intptr_t SpectraIMCalculator,
                                    eChooseRiskType choosenRiskType,
                                    const char* instrCode,
                                    const char* Code,
                                    double* arr,
                                    int32_t& pointsCount,
                                    int32_t& scenarioCount,
                                    ExtraRisksInfo* xInfo);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>chosenRiskType</b>	Тип возвращаемого риска: <ul style="list-style-type: none"> <li>• <b>0</b> - риск волатильности</li> <li>• <b>1</b> - риск волатильности и экспирации</li> </ul>	
<b>instrCode</b>	Код инструмента	
<b>Code</b>	Код клиента	Семизначный код клиента
<b>arr</b>	Зарезервированный массив	
<b>pointsCount</b>	Количество точек расчета риска	
<b>scenarioCount</b>	Количество сценариев	Число сценариев экспирации вычисляется и определяется тем, сколько раз страйк опциона укладывается между лимитами. Число сценариев определяет количество столбиков в таблице на вкладке <b>Массив рисков</b> .
<b>xInfo</b>	Дополнительная информация о массиве рисков, хранящаяся в структуре <b>ExtraRisksInfo</b> (см. раздел 3.10.14)	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94819** - функция недоступна (для клиентской версии)
- **94821** - вызов функции произведен до инициализации

### 3.10.3. GetClientFutRisksArray

Возвращает клиентские риски по фьючерсам. Функция всегда что-то возвращает, если указан корректный код фьючерса.

Функция через аргумент **arr** возвращает верхнюю часть колонки в таблице на вкладке **Массив рисков**, а через аргумент **xInfo** - нижнюю часть колонки в этой таблице.

Функция вызывается два раза:

1. Вызов с **pointsCount <= 0** возвращает в эту переменную значение - число точек расчёта риска.
2. Вызов с **pointsCount > 0** возвращает в переменную **arr** массив из **pointsCount** - точек расчёта риска. Массив должен быть предварительно зарезервирован в размере, соответствующем количеству точек расчета риска **pointsCount**, возвращённому из предыдущего вызова. Значение **pointsCount** из второго вызова не должно превышать значение из первого вызова.

```
int32_t GetClientFutRisksArray(intptr_t SpectraIMCalculator,
                              eChooseRiskType chosenRiskType,
                              const char* instrCode,
                              const char* Code,
                              double* arr,
                              int32_t& pointsCount,
                              int32_t& scenarioCount,
                              ExtraRisksInfo* xInfo);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>chosenRiskType</b>	Тип возвращаемого риска: <ul style="list-style-type: none"> <li>• <b>0</b> - риск волатильности</li> <li>• <b>1</b> - риск волатильности и экспирации</li> </ul>	
<b>instrCode</b>	Код фьючерса	
<b>Code</b>	Код клиента	Семизначный код клиента
<b>arr</b>	Зарезервированный массив	
<b>pointsCount</b>	Количество точек расчета риска	

Параметр	Описание	Примечания
<b>scenarioCount</b>	Количество сценариев	Число сценариев экспирации вычисляется и определяется тем, сколько раз страйк опциона укладывается между лимитами. Число сценариев определяет количество столбиков в таблице на вкладке <b>Массив рисков</b> .
<b>xInfo</b>	Дополнительная информация о массиве рисков, хранящаяся в структуре <b>ExtraRisksInfo</b> (см. раздел 3.10.14)	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94819** - функция недоступна (для клиентской версии)
- **94821** - вызов функции произведен до инициализации

### 3.10.4. GetClientMmsRisksArray

Возвращает клиентские риски по межмесячным спредам. Функция ничего не возвращает, если инструменты, по которым у клиента есть позиции, не входят ни в один из межмесячных спредов.

Функция через аргумент **arr** возвращает верхнюю часть колонки в таблице на вкладке **Массив рисков**, а через аргумент **xInfo** - нижнюю часть колонки в этой таблице.

Функция вызывается два раза:

1. Вызов с **pointsCount <= 0** возвращает в эту переменную значение - число точек расчёта риска.
2. Вызов с **pointsCount > 0** возвращает в переменную **arr** массив из **pointsCount** - точек расчёта риска. Массив должен быть предварительно зарезервирован в размере, соответствующем количеству точек расчета риска **pointsCount**, возвращённому из предыдущего вызова. Значение **pointsCount** из второго вызова не должно превышать значение из первого вызова.

```
int32_t GetClientMmsRisksArray(intptr_t SpectraIMCalculator,
                               eChooseRiskType choosenRiskType,
                               const char* instrCode,
                               const char* Code,
                               double* arr,
                               int32_t& pointsCount,
                               int32_t& scenarioCount,
                               ExtraRisksInfo* xInfo);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>choosenRiskType</b>	Тип возвращаемого риска: <ul style="list-style-type: none"> <li>• <b>0</b> - риск волатильности</li> <li>• <b>1</b> - риск волатильности и экспирации</li> </ul>	
<b>instrCode</b>	Код базового актива фьючерса	
<b>Code</b>	Код клиента	
<b>arr</b>	Зарезервированный массив	
<b>pointsCount</b>	Количество точек расчета риска	
<b>scenarioCount</b>	Количество сценариев	Число сценариев экспирации вычисляется и определяется тем, сколько раз страйк опциона укладывается между лимитами. Число сценариев определяет количество столбиков в таблице на вкладке <b>Массив рисков</b> .
<b>xInfo</b>	Дополнительная информация о массиве рисков, хранящаяся в структуре <b>ExtraRisksInfo</b> (см. раздел 3.10.14)	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно

- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94819** - функция недоступна (для клиентской версии)
- **94821** - вызов функции произведен до инициализации

### 3.10.5. GetClientMcsRisksArray

Возвращает клиентские риски по межконтрактным спредам. Функция ничего не возвращает, если базовые активы инструментов, по которым у клиента есть позиции, не входят ни в один из межконтрактных спредов.

Функция через аргумент **arr** возвращает верхнюю часть колонки в таблице на вкладке **Массив рисков**, а через аргумент **xInfo** - нижнюю часть колонки в этой таблице.

Функция вызывается два раза:

1. Вызов с **pointsCount <= 0** возвращает в эту переменную значение - число точек расчёта риска.
2. Вызов с **pointsCount > 0** возвращает в переменную **arr** массив из **pointsCount** - точек расчёта риска. Массив должен быть предварительно зарезервирован в размере, соответствующем количеству точек расчета риска **pointsCount**, возвращённому из предыдущего вызова. Значение **pointsCount** из второго вызова не должно превышать значение из первого вызова.

```
int32_t GetClientMcsRisksArray(intptr_t SpectraIMCalculator,
                              eChooseRiskType choosenRiskType,
                              const char* instrCode,
                              const char* Code,
                              double* arr,
                              int32_t& pointsCount,
                              int32_t& scenarioCount,
                              ExtraRisksInfo* xInfo);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>choosenRiskType</b>	Тип возвращаемого риска: <ul style="list-style-type: none"> <li>• <b>0</b> - риск волатильности</li> <li>• <b>1</b> - риск волатильности и экспирации</li> </ul>	
<b>instrCode</b>	Код спреда	
<b>Code</b>	Код клиента	
<b>arr</b>	Зарезервированный массив	
<b>pointsCount</b>	Количество точек расчета риска	
<b>scenarioCount</b>	Количество сценариев	Число сценариев экспирации вычисляется и определяется тем, сколько раз страйк опциона укладывается между лимитами. Число сценариев определяет количество столбиков в таблице на вкладке <b>Массив рисков</b> .
<b>xInfo</b>	Дополнительная информация о массиве рисков, хранящаяся в структуре <b>ExtraRisksInfo</b> (см. раздел 3.10.14)	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94819** - функция недоступна (для клиентской версии)
- **94821** - вызов функции произведен до инициализации

### 3.10.6. GetBrokerRisksArray

Возвращает риски брокера. Риски брокера (в отличие от рисков клиента) всегда представлены одним массивом.

Функция через аргумент **arr** возвращает верхнюю часть колонки в таблице на вкладке **Массив рисков**, а через аргумент **xInfo** - нижнюю часть колонки в этой таблице.

Функция вызывается два раза:

1. Вызов с **pointsCount**  $\leq 0$  возвращает в эту переменную значение - число точек расчёта риска.
2. Вызов с **pointsCount**  $> 0$  возвращает в переменную **arr** массив из **pointsCount** - точек расчёта риска. Массив должен быть предварительно зарезервирован в размере, соответствующем количеству точек расчета риска **pointsCount**, возвращённому из предыдущего вызова. Значение **pointsCount** из второго вызова не должно превышать значение из первого вызова.

Риски брокера (в отличие от рисков клиента) всегда представлены \*одним\* массивом.

```
int32_t GetBrokerRisksArray(intptr_t SpectraIMCalculator,
                           eChooseRiskType choosenRiskType,
                           const char* instrCode,
                           const char* brokerCode,
                           double* arr,
                           int32_t& pointsCount,
                           int32_t& scenarioCount,
                           ExtraRisksInfo* xInfo);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>choosenRiskType</b>	Тип возвращаемого риска: <ul style="list-style-type: none"> <li>• <b>0</b> - риск волатильности</li> <li>• <b>1</b> - риск волатильности и экспирации</li> </ul>	
<b>instrCode</b>	Код инструмента	
<b>brokerCode</b>	Код брокера	
<b>arr</b>	Зарезервированный массив	
<b>pointsCount</b>	Количество точек расчета риска	
<b>scenarioCount</b>	Количество сценариев	Число сценариев экспирации вычисляется и определяется тем, сколько раз страйк опциона укладывается между лимитами. Число сценариев определяет количество столбиков в таблице на вкладке <b>Массив рисков</b> .
<b>xInfo</b>	Дополнительная информация о массиве рисков, хранящаяся в структуре <b>ExtraRisksInfo</b> (см. раздел 3.10.14)	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94819** - функция недоступна (для клиентской версии)
- **94821** - вызов функции произведен до инициализации

### 3.10.7. GetRiskPoints

Возвращает диапазон ценовых точек, в которых вычисляются риски.

Функция вызывается два раза:

1. Вызов с **pointsCount**  $\leq 0$  возвращает в эту переменную значение - число точек расчёта риска.
2. Вызов с **pointsCount**  $> 0$  возвращает в переменную **arr** массив из **pointsCount** - точек расчёта риска. Массив должен быть предварительно зарезервирован в размере, соответствующем количеству точек расчета риска **pointsCount**, возвращённому из предыдущего вызова. Значение **pointsCount** из второго вызова не должно превышать значение из первого вызова.

```
int32_t GetRiskPoints(intptr_t SpectraIMCalculator,
                     eChooseRiskType choosenRiskType,
                     const char* instrCode,
                     bool isOption,
                     double* arr,
                     int32_t& pointsCount,
                     int32_t& scenarioCount,
                     ExtraRisksInfo* xInfo,
```



```
Enums::ClientRiskLevel level);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>chosenRiskType</b>	Тип возвращаемого риска: <ul style="list-style-type: none"> <li><b>0</b> - риск волатильности</li> <li><b>1</b> - риск волатильности и экспирации</li> </ul>	
<b>instrCode</b>	Код инструмента	
<b>isOption</b>	Тип инструмента: фьючерс или опцион.	
<b>arr</b>	Зарезервированный массив	
<b>pointsCount</b>	Количество точек расчета рисков	
<b>scenarioCount</b>	Количество сценариев	Число сценариев экспирации вычисляется и определяется тем, сколько раз страйк опциона укладывается между лимитами. Число сценариев определяет количество столбиков в таблице на вкладке <b>Массив рисков</b> .
<b>xInfo</b>	Дополнительная информация о массиве рисков, хранящаяся в структуре <b>ExtraRisksInfo</b> (см. раздел 3.10.14)	
<b>level</b>	Категория уровня риска клиента	Допустимые значения: <ul style="list-style-type: none"> <li><b>0</b> - Клиент с базовым уровнем риска (значение по умолчанию)</li> <li><b>1</b> - Клиент с Начальным Уровнем Риска (КНУР)</li> <li><b>2</b> - Клиент со Стандартным Уровнем Риска (КСУР)</li> <li><b>3</b> - Клиент с Повышенным Уровнем Риска (КПУР)</li> <li><b>4</b> - Клиент с Особым Уровнем Риска (КОУР)</li> </ul>

Возвращаемые коды ошибок (см. раздел 3.11):

- 0** - функция отработала успешно
- 94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- 94811** - ошибка входных данных
- 94819** - функция недоступна (для клиентской версии)
- 94821** - вызов функции произведен до инициализации

### 3.10.8. GetClientFinalRisksArray

Возвращает строку с итоговыми рисками, соответствующую риск-группе с указанным кодом клиента.

Риск-группой может быть:

- межконтрактный спред;
- межмесячный спред, не входящий в межконтрактный спред;
- одиночный фьючерс, не входящий в межмесячный спред.

В строку риск-группы включаются все точки, являющиеся финальными сценариями риск-группы:

- залимитные точки (ZL/ZR, UpRisk/DownRisk);
- точки лимитов концентрации с примененными коэффициентами;
- любые другие точки, используемые при взятии минимума перед сложением, для получения финального требования к ГО.

```
int32_t GetClientRiskVector(intptr_t SpectraIMCalculator,
                           eChooseRiskType chosenRiskType,
                           const char* instrCode,
                           const char* Code,
                           double* arr,
```

```
int32_t& pointsCount,
ExtraRisksInfo* xInfo);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>chosenRiskType</b>	Тип возвращаемого риска: <ul style="list-style-type: none"> <li>• <b>0</b> - риск волатильности</li> <li>• <b>1</b> - риск волатильности и экспирации</li> </ul>	
<b>instrCode</b>	Код инструмента	
<b>Code</b>	Код клиента	
<b>arr</b>	Зарезервированный массив	
<b>pointsCount</b>	Количество точек расчета риска	
<b>xInfo</b>	Дополнительная информация о массиве рисков, хранящаяся в структуре <b>ExtraRisksInfo</b> (см. раздел 3.10.14)	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94819** - функция недоступна (для клиентской версии)
- **94821** - вызов функции произведен до инициализации

### 3.10.9. GetBrokerFinalRisksArray

Возвращает строку с итоговыми рисками, соответствующую риск-группе с указанным кодом брокера.

Риск-группой может быть:

- межконтрактный спред;
- межмесячный спред, не входящий в межконтрактный спред;
- одиночный фьючерс, не входящий в межмесячный спред.

В строку риск-группы включаются все точки, являющиеся финальными сценариями риск-группы:

- залимитные точки (ZL/ZR, UpRisk/DownRisk);
- точки лимитов концентрации с примененными коэффициентами;
- любые другие точки, использующиеся при взятии минимума перед сложением, для получения финального требования к ГО.

```
int32_t GetBrokerFinalRisksArray(intptr_t SpectraIMCalculator,
                                eChooseRiskType chosenRiskType,
                                const char* instrCode,
                                const char* Code,
                                double* arr,
                                int32_t& pointsCount,
                                ExtraRisksInfo* xInfo);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>chosenRiskType</b>	Тип возвращаемого риска: <ul style="list-style-type: none"> <li>• <b>0</b> - риск волатильности</li> <li>• <b>1</b> - риск волатильности и экспирации</li> </ul>	
<b>instrCode</b>	Код инструмента	
<b>Code</b>	Код брокера	
<b>arr</b>	Зарезервированный массив	
<b>pointsCount</b>	Количество точек расчета риска	

Параметр	Описание	Примечания
<b>xInfo</b>	Дополнительная информация о массиве рисков, хранящаяся в структуре <b>ExtraRisksInfo</b> (см. раздел 3.10.14)	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94819** - функция недоступна (для клиентской версии)
- **94821** - вызов функции произведен до инициализации

### 3.10.10. GetSettlementAccountFinalRisksArray

Возвращает строку с итоговыми рисками, соответствующую риск-группе с указанным расчетным кодом.

```
int32_t GetSettlementAccountFinalRisksArray(intptr_t SpectraIMCalculator,
                                             eChooseRiskType choosenRiskType,
                                             const char* instrCode,
                                             const char* Code,
                                             double* arr,
                                             int32_t& pointsCount,
                                             ExtraRisksInfo* xInfo);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>choosenRiskType</b>	Тип возвращаемого риска: <ul style="list-style-type: none"> <li>• <b>0</b> - риск волатильности</li> <li>• <b>1</b> - риск волатильности и экспирации</li> </ul>	
<b>instrCode</b>	Код инструмента	
<b>Code</b>	Расчетный код	
<b>arr</b>	Зарезервированный массив	
<b>pointsCount</b>	Количество точек расчета риска	
<b>xInfo</b>	Дополнительная информация о массиве рисков, хранящаяся в структуре <b>ExtraRisksInfo</b> (см. раздел 3.10.14)	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94819** - функция недоступна (для клиентской версии)
- **94821** - вызов функции произведен до инициализации

### 3.10.11. GetClientFinalRisksArrayWithPositiveValue

Возвращает строку с итоговой положительной частью рисков, соответствующую риск-группе с указанным кодом клиента.

```
int32_t GetClientFinalRisksArrayWithPositiveValue (intptr_t SpectraIMCalculator,
                                                    eChooseRiskType choosenRiskType,
                                                    const char* instrCode,
                                                    const char* Code,
                                                    double* arr,
                                                    int32_t& pointsCount,
                                                    ExtraRisksInfo* xInfo);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	

Параметр	Описание	Примечания
<b>chosenRiskType</b>	Тип возвращаемого риска: <ul style="list-style-type: none"> <li>• <b>0</b> - риск волатильности</li> <li>• <b>1</b> - риск волатильности и экспирации</li> </ul>	
<b>instrCode</b>	Код инструмента	
<b>Code</b>	Код клиента	
<b>arr</b>	Зарезервированный массив	
<b>pointsCount</b>	Количество точек расчета риска	
<b>xInfo</b>	Дополнительная информация о массиве рисков, хранящаяся в структуре <b>ExtraRisksInfo</b> (см. раздел 3.10.14)	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94819** - функция недоступна (для клиентской версии)
- **94821** - вызов функции произведен до инициализации

### 3.10.12. GetBrokerFinalRisksArrayWithPositiveValue

Возвращает строку с итоговой положительной частью рисков, соответствующую риск-группе с указанным кодом брокера.

```
int32_t GetBrokerFinalRisksArrayWithPositiveValue (intptr_t SpectraIMCalculator,
                                                    eChooseRiskType chosenRiskType,
                                                    const char* instrCode,
                                                    const char* Code,
                                                    double* arr,
                                                    int32_t& pointsCount,
                                                    ExtraRisksInfo* xInfo);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>chosenRiskType</b>	Тип возвращаемого риска: <ul style="list-style-type: none"> <li>• <b>0</b> - риск волатильности</li> <li>• <b>1</b> - риск волатильности и экспирации</li> </ul>	
<b>instrCode</b>	Код инструмента	
<b>Code</b>	Код брокера	
<b>arr</b>	Зарезервированный массив	
<b>pointsCount</b>	Количество точек расчета риска	
<b>xInfo</b>	Дополнительная информация о массиве рисков, хранящаяся в структуре <b>ExtraRisksInfo</b> (см. раздел 3.10.14)	

Возвращаемые коды ошибок (см. раздел 3.11):

- **0** - функция отработала успешно
- **94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- **94811** - ошибка входных данных
- **94819** - функция недоступна (для клиентской версии)
- **94821** - вызов функции произведен до инициализации

### 3.10.13. GetSettlementAccountFinalRisksArrayWithPositiveValue

Возвращает строку с итоговой положительной частью рисков, соответствующую риск-группе с указанным расчетным кодом.

```
int32_t GetSettlementAccountFinalRisksArrayWithPositiveValue (intptr_t SpectraIMCalculator,
    eChooseRiskType choosenRiskType,
    const char* instrCode,
    const char* Code,
    double* arr,
    int32_t& pointsCount,
    ExtraRisksInfo* xInfo);
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>choosenRiskType</b>	Тип возвращаемого риска: <ul style="list-style-type: none"> <li><b>0</b> - риск волатильности</li> <li><b>1</b> - риск волатильности и экспирации</li> </ul>	
<b>instrCode</b>	Код инструмента	
<b>Code</b>	Расчетный код	
<b>arr</b>	Зарезервированный массив	
<b>pointsCount</b>	Количество точек расчета риска	
<b>xInfo</b>	Дополнительная информация о массиве рисков, хранящаяся в структуре <b>ExtraRisksInfo</b> (см. раздел 3.10.14)	

Возвращаемые коды ошибок (см. раздел 3.11):

- 0** - функция отработала успешно
- 94808** - общая ошибка, когда невозможно определить точную категорию ошибки
- 94811** - ошибка входных данных
- 94819** - функция недоступна (для клиентской версии)
- 94821** - вызов функции произведен до инициализации

### 3.10.14. Структура ExtraRisksInfo

Структура для хранения параметров, выводимых в таблице на вкладке **Массив рисков**.

```
struct ExtraRisksInfo
{
    double DownRisk;
    double ZL;
    double PnL3D;
    double PnL2D;
    double PnL2U;
    double PnL3U;
    double ZR;
    double UpRisk;
    int SubPointsCount;
};
```

Параметр	Описание	Примечания
<b>SpectraIMCalculator</b>	Handler модуля расчета ГО	
<b>DownRisk</b>	Непокрытые продажи слева	
<b>ZL</b>	Левая залимитная точка	
<b>PnL3D</b>	Значение риска концентрации третьего уровня слева	
<b>PnL2D</b>	Значение риска концентрации второго уровня слева	
<b>PnL2U</b>	Значение риска концентрации второго уровня справа	
<b>PnL3U</b>	Значение риска концентрации третьего уровня справа	
<b>ZR</b>	Правая залимитная точка	
<b>UpRisk</b>	Непокрытые продажи справа	

Параметр	Описание	Примечания
SubPointsCount	Количество элементов в массиве рисков подточек	

### 3.11. Возвращаемые коды ошибок

Код ошибки	Символьное обозначение	Описание
0	SPECTRA_IM_OK	Функция отработала успешно
94792	P2ERR_CALCGO_API_ERROR_MULTILEG_ALREADY_EXISTS	Мультилег уже существует
94808	SPECTRA_IM_COMMON_ERROR	Общая ошибка, когда невозможно определить точную категорию ошибки
94809	SPECTRA_IM_ALREADY_INITIALIZED	Попытка повторной инициализации
94810	SPECTRA_IM_END_OF_DATA	Достигнута последняя запись в массиве данных
94811	SPECTRA_IM_WRONG_ARGUMENTS	Ошибка входных данных
94812	SPECTRA_IM_NOT_ENOUGH_MEMORY	Не хватает памяти
94813	SPECTRA_IM_NOT_FOUND	Запрашиваемые данные не найдены
94814	SPECTRA_IM_STARTUP_ERROR	Ошибка связи с роутером или иная ошибка при выполнении блока STARTUP
94815	SPECTRA_IM_CONNECT_ERROR	Ошибка соединения
94816	SPECTRA_IM_LOGIN_ERROR	Ошибка при логировании пользователя
94817	SPECTRA_IM_GET_USER_INFO_ERROR	Несоответствие логина типу версии программы (клиент, фирма) или иное несоответствие в информации о пользователе
94818	SPECTRA_IM_EXCEEDED_LIMIT_OF_PARTICIPANTS	Превышение числа клиентов
94819	SPECTRA_IM_NOT_SUPPORTED	Функция для данной версии библиотеки недоступна
94820	SPECTRA_IM_NOT_CONNECTED	Нет соединения
94821	SPECTRA_IM_NOT_INITIALIZED	Вызов функции произведен до инициализации
94822	SPECTRA_IM_WRONG_PORT	Отсутствие или некорректное значение порта, указанного в ini-файле
94823	SPECTRA_IM_WRONG_MQ_TIMEOUT	Отсутствие или некорректное значение таймута, указанного в ini-файле
94995	P2ERR_SPECTRA_IM_INSTRUMENTS_ALREADY_LOADED	Повторный вызов функции GetInstrumentParameters без предварительного вызова функции очистки ClearAll