



Docs API Documentation

1/17/2025

v1.25

Revision Version	Modified Date	Revision Details
v1.0	07/30/2018	Initial Version
v1.1	11/10/2018	AgreementTransitionedFrom and AgreementTransitionedTo fields added in the agreement model on Page 12
v1.2	04/02/2019	Clearing account API added Business Dates clarified (section 5)
v1.3	04/09/2019	Added Agreement Risk calls (section 10)
v1.4	05/06/2019	No institution Id required for Rate Schedule api
v1.5	05/16/2019	Removed Executing Accounts from Rate Schedule Upload call
v1.6	06/22/2019	Added LegalEntityIdentifier to the GetParties api call
v1.7	07/08/2019	Added LegalEntityIdentifier to the GetAgreementDetails api call
v1.8	08/08/2019	Updated error response codes
v1.9	08/19/2019	Updated Enumerations
v1.10	10/01/2019	Updated Enumerations
v1.11	10/01/2019	Added generateNewApiKey call
v1.12	10/03/2019	Updated ExecutionType/Name
v1.13	11/04/2019	Updated error codes for Agreement Risk calls
v1.14	11/20/2019	Fixed 404 error and ClearingAccountType example value
v1.15	02/20/2020	Action sequence customizations in UI
v1.16	06/15/2020	Added UnsupportedProducts to GetAgreementDetails call
v1.17	08/19/2020	Added Rest Api Versioning Principles
V1.18	11/26/2020	Clarified Expiration date for CA data in agreement details
V1.19	12/16/2020	Added IsActive flag on Rate row of Products Included indication of the future v2 of the APIs.
V1.20	02/18/2021	Fixed new rate type mentions
V1.21	11/28/2022	Replaced v1 calls with v2 Added Expanded Agreement Details and Expanded Rate Schedule API calls Updated enumerations to match data in v2 calls
V1.22	02/29/2024	Added tags for legal language change indicators in Agreement Details calls
V1.23	03/01/2024	Added Agreement Language API call to Agreements section
V1.24	05/28/2024	Updated links to reference data API calls



V1.25	01/17/2025	Revised to include additional API calls – Put Agreement History, Put Agreement Comments, Put Agreement Workstream Status Updated incorrect references in Agreement API calls
--------------	------------	--



1. Introduction	4
2. Docs API developer testing page	4
3. Acquiring an API account and key	5
4. REST API Versioning Principles	6
5. Primary list of API Response Fields	7
6. Business Dates	12
7. Enumeration types returned from the responses	12
8. Common Error Messages	18
9. Agreement	18
a. Get Agreement Details	18
b. Get Expanded Agreement Details	24
c. Get Updated Agreements	28
d. Get Agreement Language	30
e. Put Agreement Comments	34
f. Put Agreement History	36
g. Put Agreement Workstream Status	38
10. Clearing Accounts	41
a. Add clearing account	41
b. Add multiple clearing accounts	42
c. Delete clearing account	44
d. Expire clearing account	44
11. Agreement Risk	45
a. Get Agreement Risk Details	45
b. Get Agreement Risk Updates	47
c. Set Limits	48
d. Delete Limits	50
e. Set Conditions	51
f. Delete Conditions	52
g. Set Product Exclusions	54
h. Delete Product Exclusions	55
12. Rate Schedule	56
a. Upload Rate Schedule	56



b.	Get Updated Rate Schedules	60
c.	Download Rate Schedule	60
d.	Download Expanded Rate Schedule	62
e.	Delete Rate Schedule.....	65
13.	Reference Data	65
a.	Get Parties	65
b.	Get Exchanges.....	67
c.	Get Exchange Firms	67
d.	Get Exchange Product Groups	68
e.	Get Enumerations	70
14.	User	74
a.	Generate New Api Key.....	74

1. Introduction

This guide provides information about the DOCS REST API. Using this document, you can find the process to generate the API key, the various API actions/calls and their request and response information.

2. Docs API developer testing page

The Docs API developer testing page (screenshot below) is available at:

- PSIM (Prod Simulation): <https://psim.fia-tech.com/DocsApi>
- Production: <https://myapps.fia-tech.com/DocsApi>.

On this page, any authenticated API user can send requests and receive responses back using an [API key](#) entitled to see data. It is a good testing tool for quick API example calls.

Api Key <input type="text"/>		Api Documentation
Agreement Show/Hide List Operations Expand Operations		
GET	/agreement/updates	Get Updated Agreements based on the fromDate and toDate in the provided timezone
GET	/agreement/details	Get Agreement Details based on the passed AgreementId
Clearing Accounts Show/Hide List Operations Expand Operations		
DELETE	/agreement/clearingAccount	Deletes clearing account from an agreement
POST	/agreement/clearingAccount	Adds a new clearing account to an agreement
POST	/agreement/clearingAccounts	Adds multiple new clearing accounts to an agreement
PUT	/agreement/expireClearingAccount	Updates the expiration date on a clearing account
Agreement Risk Show/Hide List Operations Expand Operations		
GET	/agreementRisk/updates	Get Agreement Risk updates based on the fromDate and toDate in the provided timezone
GET	/agreementRisk/details	Get Agreement Risk Limits and Conditions based on the passed AgreementId
DELETE	/agreementRisk/limits	Delete Agreement Risk Limits based on the passed AgreementId
POST	/agreementRisk/limits	Set limits on an agreement
DELETE	/agreementRisk/conditions	Delete Agreement Risk Conditions based on the passed AgreementId
POST	/agreementRisk/conditions	Set conditions on an agreement
DELETE	/agreementRisk/productExclusions	Delete product exclusions on agreement
POST	/agreementRisk/productExclusions	Set product exclusions on an agreement

Rate Schedule		Show/Hide	List Operations	Expand Operations
GET	/rateSchedule/updates			Get Updated Rate Schedules
GET	/rateSchedule/download			Download Rate Schedule
POST	/rateSchedule/upload			Upload Rate Schedule
DELETE	/rateSchedule/delete			Delete Rate Schedule
Reference Data		Show/Hide	List Operations	Expand Operations
PUT	/referenceData/parties			Get Parties
PUT	/referenceData/exchanges			Get Exchanges
PUT	/referenceData/exchangeFirms			Get Exchange Firms
PUT	/referenceData/exchangeProductGroups			Get Exchange Product Groups
GET	/referenceData/enumerations			Get Enumerations
User		Show/Hide	List Operations	Expand Operations
GET	/user/generateNewApiKey			Generates a new Api Key

3. Acquiring an API account and key

The new Docs API is built using REST (Representational State Transfer). To gain access to this API, you will need a user account entitled to access the API, as well as an API token/key. This token is your identity when you call the API with the token in the request header. You do not need to send a user ID/password pair when accessing the API.

None of the API calls can be done through a browser address bar even if it is only a GET call since an API key must be present in the header of the request. An API tool like curl, postman or fiddler etc. is required to test the URL. Alternatively, calls can also be tested using the testing page hosted in the application. For more information, see the previous section, [Docs Api developer testing page](#).

API tokens can be generated by properly entitled user accounts by the user themselves, or FIA Tech. See the following section for additional information.

How to get a user account entitled to API access

To setup an account to access the API please call FIA Tech Client Services at 202-772-3000 or email at docs.support@fia-tech.com.

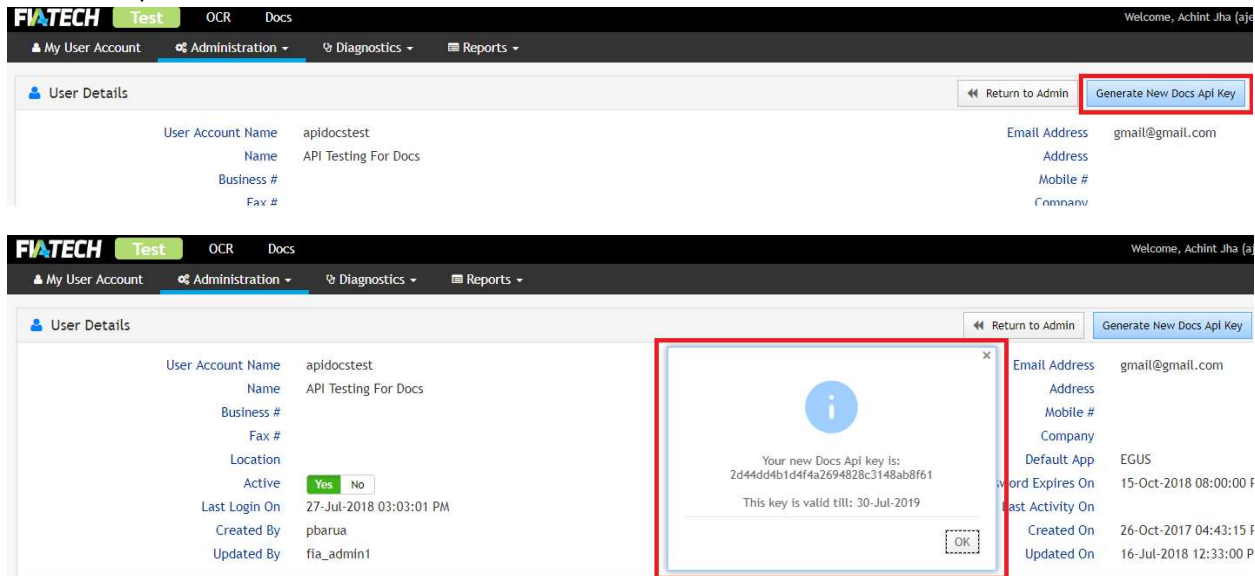
How to generate a Docs API token

The API key needs to be copied manually by the API user on their side and used in the API calls. If the user loses the key, a new one will need to be generated.

A Docs API key can be generated by any one of the following ways:

1. **Through UserHub – Account Management:**
 2. Login to UserHub using the API user account Id and password
 3. Click on Welcome, <Api user name>

4. Click on the button “Generate New Docs Api Key”
5. A message box will be displayed which will show the API Key. The key needs to be kept or copied at a safe place to be used in API calls. If the key is lost, it can only be created again following steps 1 onwards.



6. **Call FIA Tech Client Services** to provide you a token. They can only provide you if you are already set up as a valid API user.
7. **Through Docs Api (only if you already have a valid key):** If you already have a valid api key and want to change it, you may use the “generateNewApiKey” method of the Docs Api. See “[User](#)” section for details.

4. REST API Versioning Principles

REST APIs are flexible in that they allow less rigorous specifications for the public interface when compared to other technologies. However, it is important to be specific about what approach FIA Tech uses when versioning an API so that breaking changes are not introduced into client systems.

FIA Tech rules for API Versioning:

We define a ‘breaking change’ to be any change which has one or more of the following criteria:

1. Adding a new or modifying an existing validation to an existing resource.
2. Requiring a parameter that was not required before.
3. Changing existing error response codes/messages.
4. Modifying the expected payload of webhooks and async callbacks.
5. Changing the data type of an existing field.
6. Renaming a field or endpoint.

7. Adding a new feature that will change the meaning of a field.
8. Removing an existing field or endpoint.
9. Changing the URL structure of an existing endpoint.
10. Adding or removing any fixed enumeration in a field.

Any breaking change to an API will result in a new version of that API being published:

1. A new version of an API will include the version number in the URL.
2. Firms can individually decide when to move their systems to consume the new version within a specified time frame.
3. Old versions must have a clear and fixed expiration date after which they are decommissioned.

Any release that contains only non-breaking changes can be expected to be released into production with suitable notice but without industry coordination. Examples of non-breaking changes are:

1. Adding a new method.
2. Adding an optional parameter to an existing method.
3. Adding new nodes or attributes to existing JSON results.

5. Primary list of API Response Fields

This section outlines the primary list of response/request fields that appear across multiple API calls. They are listed with a brief description. Note that these fields are listed here in alphabetical order, not the order they appear in the API response.

Bands

Field	Description
ApplyTo	Indicates whether this highest band rate should be applied to all contracts, or if all preceding bands should be used in calculation. One of the possible values from RateApplyTo enumeration
BandNum	Sequence number of this Band
BandRateType	One of the possible values from RateType for this band
BandValueType	One of the possible values from BandValueType
End	The end value for this band based on BandValueType
Rate	The rate applied when this band is matched
Start	The start value for this Band based on BandValueType

Clearing Account

Field	Description
AccountId	Clearing account ID
ClearingAccountType	Type of clearing account
Description	Description of the account
EffectiveDate	Effective Date for clearing account (this is a business date)

ExpirationDate	Expiration Date for clearing account (this is a business date). If this clearing account is on an agreement and there is no expiration date, then the agreement termination date is used as the Expiration Date.
-----------------------	--

Comments

Field	Description
CreatedOn	Date comment was created
CreatedBy	The user who created the comment. See CreatedBy
Workstream	Optional workstream to which the comment is directed
ToParty	Optional party on the agreement to which the comment is directed
CommentType	The type of comment that was posted to the agreement.
CommentDetails	Text of comment

Created By

Field	Description
UserName	System created username
UserEmail	Email address associated with user account
UserParty	Party on behalf of which the user is acting

Exchange

Field	Description
DirectBill	Indicates if this exchange is to be billed directly. Values: true/false
ExchangeCode	Indicates code for the exchange
Rates	List of Rates defined for this exchange
CreatedOn	Indicates created on datetimestamp
UpdatedOn	Indicates updated on datetimestamp

Executing Account

Field	Description
AccountId	Executing account Id
Description	Description of the account
EffectiveDate	Effective Date for executing account (this is a business date)
ExpirationDate	Expiration Date for executing account (this is a business date)

History Records

Field	Description
CreatedOn	Date history record was created
CreatedBy	The user who took the action. See CreatedBy

ActionDetails	Description of action taken on agreement
---------------	--

Legal Language Paragraphs

Field	Description
Order	Placement of the paragraph in the specific agreement. Starts at 1
HasAgreementTextChanges	True/False indicator if the paragraph has been changed from standard. Set to false where the language is unchanged
Text	Text of the paragraph

Party

Field	Description
AgreementPartyRole	The role for this party on the agreement. (This field is returned only if the Party is being returned as a result of a call to the Agreement API)
InstitutionId	InstitutionId of the party. If the party is not a sub-party, this will be identical to the ParentId
IsActive	True/false indicator of whether or not the party is active in Docs and can be added to new agreements. Set to true if the party is active
IsConformed	True/false indicator of whether or not the party is subscribed to Docs. Set to true if the party is conformed (subscribed)
Name	Name of the party
ParentId	Parent PartyId of the party. If the party is not a sub-party, this will be identical to the InstitutionId
PartyFirms	List of Party Firms in associated to the party
PartyId	Unique identifier for the party
LegalEntityIdentifier	LEI for the party
CreatedOn	Date when the party was created
UpdatedOn	Date when the party was last updated

Party Firm

Field	Description
ExchangeCode	Exchange Code the FirmCode applies for
FirmCode	Exchange Firm Code for the Party
PrimaryFlag	Indicates if the Exchange Firm Code is primary (Y/N)
CreatedOn	Date when the partyfirm was created
UpdatedOn	Date when the partyfirm was last updated

Product

Field	Description
ProductCode	Product Code
ProductName	Product Name

ProductType	Type of instrument
CreatedOn	Date when the product was created
UpdatedOn	Date when the product was last updated

Product Group

Field	Description
ProductGroupCode	Product group code for a Products
Products	List of Products under this product group

Rate

Field	Description
AddOnAmount	Indicates Add On amount.
AddOnApply	Indicates Add On Apply Type (Per Contract or Per TradeType)
Amount	Indicates Rate Amount.
Bands	List of Bands involved in the Rates
BbgExchangeCode	Bloomberg Exchange Code associated with the product , if available. Only included in the Expanded calls
BbgRootCode	Bloomberg Root Code associated with the product , if available. Only included in the Expanded calls
BbgYellowKey	Bloomberg Yellow Key associated with the product , if available. Only included in the Expanded calls
DefaultExecutionType	Indicates if the ExecutionType is default (Y/N)
ExecutionType	Indicates the code of the execution type that applies to the rate
ExecutionTypeName	Indicates execution type name
GmiExchangeCode	GMI Exchange Code associated with the product , if available. Only included in the Expanded calls
GmiContractType	GMI Contract Type associated with the product , if available. Only included in the Expanded calls
GmiFuturesCode	GMI Futures Code associated with the product , if available. Only included in the Expanded calls
IsActive	Indicates the status of each product as either active or inactive
Max	Maximum amount for the Rate.
Min	Minimum amount for the Rate.
MinMaxApply	Indicates Min Max Apply Type. One of the possible values from RateApply enumeration.
MinMaxDefine	Indicates Min Max Define type. Type of value in the Min and Max fields
OpMic	Indicates the operating MIC of the product , if available
ProductCode	Indicates product code
ProductGroupCode	Indicates product group code
ProductType	Indicates product type
Rank	Indicates order in which to process the rate within the rate schedule for trade matching

RateCurrency	3-digit standard ISO currency code (for e.g. EUR, USD etc)
RateType	Indicates the type of this Rate
SegMic	Indicates the segment MIC of the product , if available
SpreadType	Indicates the Spread Type of this Rate
Tiers	List of Tiers involved in the Rates. Returned only if rate type is Vol Disc. If it's not Vol Disc, this is not returned
TradeType	Indicate the Trade Type of this Rate

Rate Schedule

Field	Description
Exchanges	List of Exchanges on the Rate Schedule
ExecutingAccounts	List of Executing Accounts available on this Rate Schedule
FileName	Name of the Rate Schedule

Tier

Field	Description
Amount	Indicates Rate Amount for this tier
ApplyTo	Indicates whether the tier applies to all activity for the period or Next activity for the period. One of the possible values from RateApplyTo enumeration
Period	For Vol Disc rates, the period for which the tier applies.
StartType	For Vol Disc rates, one of the StartType of tier for the rate.
StartTypeValue	For Vol Disc rates, the start number or amount of each tier

Risk Limit

Field	Description
Type	Risk Limit Type Enum
Currency	Indicates currency code of limit amount
Amount	Indicates the limit amount
LimitEffectiveDate	Effective date of limit
CreatedOn	Creation date of limit
Version	Version number
State	Risk Limit State Enum

Risk Condition

Field	Description
Type	Risk Condition Type Enum
Details	Condition details in case of custom condition
CreatedOn	Creation date of condition

Risk Product Exclusion

Field	Description
Region	Exchange region name
ProductName	Product Name
EffectiveDate	Effective date of product exclusion
ExchangeCode	Exchange code
ProductGroupCode	Product group code
ProductCode	Product code
ProductType	Product type

Unsupported Products

Field	Description
ExchangeCode	Exchange code
ProductGroupCode	Product group code
ProductCode	Product code
ProductType	Product type
UnsupportedBy	List of party roles that unsupported the product
RateScheduleFileNames	List of Rate Schedules which contain this product
HasUnsupportedRateSchedules	At least one Rate Schedule out of RateScheduleFileNames have Unsupported party roles

Workstream Approvals

Field	Description
Approved	True/false flag indicating if the workstream is approved
ApprovedBy	User who approved the workstream
ApprovedOn	Date/time when the workstream is approved
Type	Type of workstream that was approved

6. Business Dates

Business Dates (date only without a time component) on all the responses are:

- EffectiveDate
- ExpirationDate
- TerminationDate

7. Enumeration types returned from the responses

AgreementPartyRole	AdditionalTrader
	Trader
	ExecutionBroker

	ClearingBroker
	Customer
	ExecutingAdministrativeClearer
	ClearingAdministrativeClearer
	CarryingBroker
	OrderPassingBroker
	LMEClearingMember
	LMEExecutingMember
	Broker

AgreementState	Pending
	Draft
	PendingInitiation
	InitiationRejected
	Composition
	Execution
	Executed
	Terminated
	Stale
	Deleted
	CompositionLocked

AgreementType	STDC
	STDT
	STDO
	STDCO
	EFPT
	EFPC
	EFPBT
	EFPBC
	EFPBDT
	EFPBDC
	GUSA
	CLSA
	LMECCB
	LMEEXM
	LMEEXMCB
	LMECEAC
	LMECCAC
	LMECEACCAC
	LMECEACCB
	LMECEACCACCB
	LMETCB
	LMETXM

	LMETXMCB
	LMETEAC
	LMETCAC
	LMETEACCAC
	LMETEACCB
	LMETEACCACCB
	LMEC
	LMET

BrrdVersion	EU
	Hybrid
	UK
	None

BandValueType	BPS
	Notional
	NumberOfContracts
	Premium
	TradePrice

ClearingAccountType	CustomerClearingAccount
	FcmProcessingAccount

CommentType	Comment
	Rejection
	PartyNotFound
	Waiver

ExecutionType	ExecutionType	ExecutionTypeName
	Y	Electronic
	Z	ClientPit
	W	Desk
	X	DeskPit
	A	PhoneSimple
	B	PhoneComplex
	LC	FcmProvidedScreen
	C	VendorProvidedPlatformBilledByExecutingBroker
	LD	OtherProvidedScreen
	D	OtherIncludingOtherProvidedScreen
	E	ClientProvidedPlatformControlledByFcm
	H	PremiumAlgorithmicTradingProviderBilledByExecutingBroker

F	ClientProvidedPlatformDirectToExchange
LG	FcmApiOrFix
G	SponsoredAccessViaExchangeApiOrFixProvidedByExecutingBroker
LH	AlgoEngine
J	PriceAtExecution
K	K
M	M
N	N
P	P
Q	Q
R	R
S	S
T	T
U	U
V	V
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9

MinMaxDefine	Amount
	PercentPremium
	PercentNotional

PartyRole	Trader
	ExecutionBroker
	ClearingBroker
	Customer

Period	Daily
	Monthly

ProductType	COMBO
	Forwrd
	FUT
	Future
	Mixed

	OFF
	OOO
	OOF
	OPT
	Option
	PHY
	Spot
	Swap
	Swaps

RateApply	PerContract
	PerTrade

RateApplyTo	UseOnlyOne
	UseAllThatApply

RateType	PerContract
	PerTrade
	VolDisc
	PercentPremium
	PercentNotional
	UnderlierPriceBPS
	TradePriceBPS
	MWH
	BTU
	MBTU
	MMBTU
	TONNES
	Barrels
	MetricTons
	PerOunce
	PerTherm
	Vega
	PercentDividendNotional
	Band

SpreadType	Spread
	PitSpread
	Roll
	Hedge
	Fly
	Straddle

	Strangle
	Switch
	TAS
	Condor

StartType	Amount
	Number
	PremiumPrice
	NotionalPrice
	NumberOfContracts
	NumberOfTrades

TradeType	BlockTrade
	ExchangeForPhysical
	ExchangeForRisk
	ExchangeForSwap
	BasisTrade
	PerAllocation

RiskStatus	NoLimitsAndConditions
	HasLimitsAndConditions
	RestrictedDueToCustomCondition
	RestrictedDueToTradeByTrade
	Suspended

RiskLimitType	DailyInitialMargin
	DailyMaxContracts

RiskLimitState	Draft
	Approved

RiskConditionType	AgreementSuspended
	TradeByTradeReview
	Custom

WorkflowType	Billing
	Regular
	Uploaded

8. Common Error Messages

For instances where there is a problem with a Request or Response, the API will return a HTTP Response code as below, along with an error JSON.

- 500 for internal errors
- 400 for all types of validation errors
- 404 for resource not found errors (like Agreement Not Found)

Below are some common error messages to any API call. Individual API calls documents the error JSON specific to each call.

```
{
  "ErrorCode": "InternalError",
  "ErrorMessage": "Internal Error"
}
```

```
{
  "ErrorCode": "InvalidTimeComponent",
  "ErrorMessage": "Date Time value is missing timezone component"
}
```

```
{
  "ErrorCode": "InvalidDateRange",
  "ErrorMessage": "'From Date' cannot be after 'To Date'"
}
```

9. Agreement

a. Get Agreement Details

The Agreement Details API allows you to request agreement details based on API user's party entitlements. This API returns agreement details for all agreement states including "Deleted". Within this response, agreement rate schedule details are returned as they exist on the agreement. This means that groups appear as is and are not broken out into the underlying products.

If you wish to retrieve the full agreement details with all underlying products and any subscribed symbologies, see the [Expanded Agreement Details API](#) call section.

Request

```
GET /AgreementApi/v2/agreement/<AgreementId>
```

Field	Description
AgreementId	Unique identifier for the Agreement for which details will be retrieved

Response

Field	Description
AgreementId	Unique identifier for an Agreement
AgreementTransitionedFrom	The agreement Id from which it was cloned (may be empty if this agreement was not part of a give-up agreement transition)
AgreementTransitionedTo	The agreement Id to which it was cloned (may be empty if this agreement was not part of a give-up agreement transition)
AgreementType	Type of the agreement
BillWho	Party to which you can bill for this agreement (except Executing Broker)
BRRDVersion	Indicates which BRRDversion of FIA's standard language has been added to the agreement
ClearingAccounts	List of ClearingAccounts in the agreement
CreatedOn	Date Agreement was created
CustomerReference	Name of Customer party for which the details are being downloaded. If there is no customer party on the agreement, then the Trader party name is returned
EffectiveDate	Date on which Agreement becomes effective (this is a business date)
ExecutedDate	Executed date time stamp of the agreement in UTC
FollowingAgreementId	Agreement Id to which it was amended (may be empty if this agreement was not amended)
GroupId	Unique identifier for the agreements which are related
HasAgreementTextChanges	Indicates whether or not the agreement text has been modified from the standard. Will be true for agreements with changes, false for all others
InitiatedOn	Initiated date time stamp in UTC
Jurisdiction	Agreement jurisdiction
LastModified	Last modified (touched) date time stamp in UTC
Paragraph10Addition	Indicates if the agreement has added FIA's standard paragraph 10 language for UK governing law. Will be true for agreements with changes, false for all others
Parties	List of Parties involved in the agreement
PriorAgreementId	Agreement Id from which it was amended (may be empty if this agreement is not an amendment)
ProfitCenter	Profit Center for this agreement. This is a client defined value, it is not validated by Docs.
RateSchedules	List of RateSchedules in the agreement
ReferencedAgreement	Agreement ID of the referenced Give-Up Agreement if the agreement is of workflow type "Billing"
ReferencedAgreementEffectiveDate	Effective Date of the agreement listed as the Referenced Agreement
State	State of the agreement

Supplemental Doc Count	Count of supplemental docs on the agreement
TemplateVersion	Year of the agreement template version
TerminationDate	Date on which agreement becomes terminated (this is a business date)
UnsupportedProducts	List of UnsupportedProducts in the agreement
WorkflowType	Type of Workflow applied to the agreement

Response JSON example

```
{
  "CreatedOn": "2023-10-25T23:34:29.793Z",
  "ExecutedDate": "2024-01-04T02:45:30.413Z",
  "CustomerReference": "Atlantic yellow-nosed albatross",
  "ProfitCenter": null,
  "SuppDocCount": 0,
  "GroupId": 41520,
  "PriorAgreementId": null,
  "FollowingAgreementId": null,
  "AgreementTransitionedFrom": null,
  "AgreementTransitionedTo": null,
  "BillWho": "ClearingBroker",
  "Jurisdiction": "England and Wales",
  "HasAgreementTextChanges": true,
  "Paragraph10Addition": false,
  "BrrdVersion": null,
  "AgreementId": 41520,
  "AgreementType": "STDT",
  "TemplateVersion": "2017",
  "State": "Executed",
  "InitiatedOn": "2023-10-25T23:34:29.887Z",
  "LastModified": "2024-01-04T02:45:30.41Z",
  "EffectiveDate": "2024-01-04",
  "TerminationDate": null,
  "WorkflowType": "Regular",
  "ReferencedAgreement": null,
  "ReferencedAgreementEffectiveDate": null,
  "Parties": [
    {
      "AgreementPartyRole": "ClearingBroker",
      "PartyFirms": [],
      "IsConformed": true,
      "PartyId": 100113,
      "Name": "Gringotts Wizarding Bank",
      "ParentId": 0,
      "InstitutionId": 0,
      "IsActive": true,
      "LegalEntityIdentifier": null,
      "CreatedOn": "2018-07-25T20:23:05.166Z",
      "UpdatedOn": "2024-01-10T15:42:26.41Z"
    },
    {
      "AgreementPartyRole": "ExecutionBroker",
      "PartyFirms": [],

```

```

    "IsConformed": false,
    "PartyId": 100383,
    "Name": "TestpartyNonConforming",
    "ParentId": 0,
    "InstitutionId": 0,
    "IsActive": true,
    "LegalEntityIdentifier": null,
    "CreatedOn": "2019-01-30T16:47:02.876Z",
    "UpdatedOn": "2023-10-24T09:56:54.966Z"
  },
  {
    "AgreementPartyRole": "Trader",
    "PartyFirms": [],
    "IsConformed": false,
    "PartyId": 111394,
    "Name": "Atlantic yellow-nosed albatross",
    "ParentId": 100476,
    "InstitutionId": 100476,
    "IsActive": true,
    "LegalEntityIdentifier": null,
    "CreatedOn": "2023-05-16T18:24:14.576Z",
    "UpdatedOn": "2023-05-16T18:24:14.683Z"
  }
],
"ClearingAccounts": [
  {
    "ClearingAccountType": "CustomerClearingAccount",
    "AccountId": "154864158",
    "Description": null,
    "EffectiveDate": "2024-01-04",
    "ExpirationDate": null
  },
  {
    "ClearingAccountType": "CustomerClearingAccount",
    "AccountId": "1584487941",
    "Description": null,
    "EffectiveDate": "2024-01-04",
    "ExpirationDate": null
  },
  {
    "ClearingAccountType": "CustomerClearingAccount",
    "AccountId": "22225588",
    "Description": null,
    "EffectiveDate": "2024-01-04",
    "ExpirationDate": null
  },
  {
    "ClearingAccountType": "CustomerClearingAccount",
    "AccountId": "dddd",
    "Description": null,
    "EffectiveDate": "2024-01-04",
    "ExpirationDate": null
  }
],
"RateSchedules": [

```

```

{
  "ExecutingAccounts": [],
  "Exchanges": [
    {
      "Rates": [
        {
          "ProductGroupCode": "All",
          "ProductCode": "All",
          "ProductType": "Mixed",
          "TradeType": null,
          "SpreadType": null,
          "RateType": "PerContract",
          "ExecutionType": "Z",
          "ExecutionTypeName": "ClientPit",
          "Amount": 0.4,
          "Min": null,
          "Max": null,
          "MinMaxDefine": null,
          "MinMaxApply": null,
          "AddOnAmount": null,
          "AddOnApply": null,
          "RateCurrency": "BRL",
          "DefaultExecutionType": true,
          "Tiers": [],
          "Bands": [],
          "IsActive": true
        },
        {
          "ProductGroupCode": "All",
          "ProductCode": "All",
          "ProductType": "Mixed",
          "TradeType": null,
          "SpreadType": null,
          "RateType": "PerContract",
          "ExecutionType": "Y",
          "ExecutionTypeName": "Electronic",
          "Amount": 0.3,
          "Min": null,
          "Max": null,
          "MinMaxDefine": null,
          "MinMaxApply": null,
          "AddOnAmount": null,
          "AddOnApply": null,
          "RateCurrency": "BRL",
          "DefaultExecutionType": false,
          "Tiers": [],
          "Bands": [],
          "IsActive": true
        }
      ],
      "ExchangeCode": "BSE",
      "DirectBill": false
    }
  ],
  "RateScheduleId": 12925,

```

```

    "FileName": "Test with Banded Rates",
    "DownloadFileName": null,
    "CreatedOn": "2020-02-11T18:57:45.676Z",
    "UpdatedOn": "2024-02-26T19:28:51.166Z"
  }
],
"UnsupportedProducts": [],

```

Response JSON example for a **DELETED** agreement

```

{
  "CreatedOn": "2023-05-16T18:55:49.293Z",
  "ExecutedDate": null,
  "CustomerReference": null,
  "ProfitCenter": null,
  "SuppDocCount": 0,
  "GroupId": 0,
  "PriorAgreementId": null,
  "FollowingAgreementId": null,
  "AgreementTransitionedFrom": null,
  "AgreementTransitionedTo": null,
  "BillWho": null,
  "Jurisdiction": null,
  "Parties": null,
  "ClearingAccounts": null,
  "RateSchedules": null,
  "UnsupportedProducts": null,
  "HasAgreementTextChanges": false,
  "Paragraph10Addition": false,
  "BrrdVersion": null,
  "AgreementId": 40915,
  "AgreementType": null,
  "TemplateVersion": null,
  "State": "Deleted",
  "InitiatedOn": null,
  "LastModified": "2023-08-28T21:00:30.38Z",
  "EffectiveDate": null,
  "TerminationDate": null,
  "WorkflowType": null,
  "ReferencedAgreement": null,
  "ReferencedAgreementEffectiveDate": null,
}

```

Error Response JSON example

```

{
  "ErrorCode": "InvalidAgreementId",
  "ErrorMessage": "Agreement Id is not valid"
}

```

```

{

```

```

"ErrorCode": "AgreementNotFound",
"ErrorMessage": "Agreement not found"
}

```

b. Get Expanded Agreement Details

The Expanded Agreement Details call includes all the information of the Agreement Details call plus the rate schedule broken out into individual products and enriched with cross-reference product information.

The call can be used in two manners. Either the symbology parameters can all be left unspecified, in which case the response will return to the user all symbolologies to which the firm is subscribed, or a specific set of symbolologies can be specified. To include a specific symbology, the value for the requested data should be set to “true”, otherwise it should be set as “false”.

Regardless of which symbolologies are specified, exchange product information along with segment and operating mic will always be included for all products.

Request

GET /AgreementApi/v2/agreement/expanded/<AgreementId>

Field	Description
AgreementId	Unique identifier of the agreement for which details are to be fetched
Bbg	Indication to include bbg symbolologies in response
Gmi	Indication to include gmi symbolologies in response

Response

Field	Description
AgreementId	Unique identifier for an Agreement
AgreementTransitionedFrom	The agreement Id from which it was cloned (may be empty if this agreement was not part of a give-up agreement transition)
AgreementTransitionedTo	The agreement Id to which it was cloned (may be empty if this agreement was not part of a give-up agreement transition)
AgreementType	Type of the agreement
BillWho	Party to which you can bill for this agreement (except Executing Broker)
BRRDVersion	Indicates which BRRDversion of FIA’s standard language has been added to the agreement
ClearingAccounts	List of ClearingAccounts in the agreement
CreatedOn	Date Agreement was created
CustomerReference	Name of Customer party for which the details are being downloaded. If there is no customer party on the agreement, then the Trader party name is returned

EffectiveDate	Date on which Agreement becomes effective (this is a business date)
ExecutedDate	Executed date time stamp of the agreement in UTC
FollowingAgreementId	Agreement Id to which it was amended (may be empty if this agreement was not amended)
GroupId	Unique identifier for the agreements which are related
HasAgreementTextChanges	Indicates whether or not the agreement text has been modified from the standard. Will be true for agreements with changes, false for all others
InitiatedOn	Initiated date time stamp in UTC
Jurisdiction	Agreement jurisdiction
LastModified	Last modified (touched) date time stamp in UTC
Paragraph10Addition	Indicates if the agreement has added paragraph 10 language for UK governing law. Will be true for agreements with changes, false for all others
Parties	List of Parties involved in the agreement
PriorAgreementId	Agreement Id from which it was amended (may be empty if this agreement is not an amendment)
ProfitCenter	Profit Center for this agreement. This is a client defined value, it is not validated by Docs.
RateSchedules	List of RateSchedules in the agreement
ReferencedAgreement	Agreement ID of the referenced Give-Up Agreement if the agreement is of workflow type "Billing"
ReferencedAgreementEffectiveDate	Effective Date of the agreement listed as the Referenced Agreement
State	State of the agreement
Supplemental Doc Count	Count of supplemental docs on the agreement
TemplateVersion	Year of the agreement template version
TerminationDate	Date on which agreement becomes terminated (this is a business date)
UnsupportedProducts	List of UnsupportedProducts in the agreement
WorkflowType	Type of Workflow applied to the agreement

Response JSON example

```
{
  "CreatedOn": "2024-02-20T14:03:21.677Z",
  "ExecutedDate": null,
  "CustomerReference": "Citrine Wagtail",
  "ProfitCenter": null,
  "SuppDocCount": 0,
  "GroupId": 52055,
  "PriorAgreementId": null,
  "FollowingAgreementId": null,
  "AgreementTransitionedFrom": null,
  "AgreementTransitionedTo": null,
  "BillWho": "ClearingBroker",
}
```

```

"Jurisdiction": "The Jungle",
"HasAgreementTextChanges": false,
"Paragraph10Addition": false,
"BrrdVersion": null,
"AgreementId": 52055,
"AgreementType": "STDT",
"TemplateVersion": "2017",
"State": "Execution",
"InitiatedOn": "2024-02-20T14:03:21.78Z",
"LastModified": "2024-02-29T15:46:21.897Z",
"EffectiveDate": null,
"TerminationDate": null,
"WorkflowType": "Regular",
"ReferencedAgreement": null,
"ReferencedAgreementEffectiveDate": null,
"Parties": [
  {
    "AgreementPartyRole": "ExecutionBroker",
    "PartyFirms": [],
    "IsConformed": false,
    "PartyId": 100132,
    "Name": "Clampett Oil",
    "ParentId": 0,
    "InstitutionId": 0,
    "IsActive": true,
    "LegalEntityIdentifier": null,
    "CreatedOn": "2018-07-25T20:23:05.166Z",
    "UpdatedOn": "2021-12-22T12:16:27.846Z"
  },
  {
    "AgreementPartyRole": "ClearingBroker",
    "PartyFirms": [],
    "IsConformed": true,
    "PartyId": 100356,
    "Name": "Gringotts Wizarding Bank - EU division",
    "ParentId": 100113,
    "InstitutionId": 100113,
    "IsActive": true,
    "LegalEntityIdentifier": null,
    "CreatedOn": "2018-09-18T14:56:19.72Z",
    "UpdatedOn": "2023-12-13T08:53:35.69Z"
  },
  {
    "AgreementPartyRole": "Trader",
    "PartyFirms": [],
    "IsConformed": false,
    "PartyId": 100480,
    "Name": "Citrine Wagtail",
    "ParentId": 100476,
    "InstitutionId": 100476,
    "IsActive": true,
    "LegalEntityIdentifier": null,
    "CreatedOn": "2020-03-10T16:39:38.63Z",
    "UpdatedOn": "2020-03-10T16:39:38.63Z"
  }
]

```

```

],
"ClearingAccounts": [
  {
    "ClearingAccountType": "CustomerClearingAccount",
    "AccountId": "1367802",
    "Description": null,
    "EffectiveDate": null,
    "ExpirationDate": null
  },
],
"RateSchedules": [
  {
    "ExecutingAccounts": [],
    "Exchanges": [
      {
        "Rates": [
          {
            "OpMic": "XASX",
            "SegMic": "XSFE",
            "BbgExchangeCode": "SFE",
            "BbgYellowKey": "Comdty",
            "BbgRootCode": "SX",
            "GmiExchangeCode": "18",
            "GmiContractType": "",
            "GmiFuturesCode": "US",
            "ProductGroupCode": "Ag-AUD",
            "ProductCode": "US",
            "ProductType": "Future",
            "TradeType": "BlockTrade",
            "SpreadType": "Hedge",
            "RateType": "TradePriceBPS",
            "ExecutionType": "W",
            "ExecutionTypeName": "Desk",
            "Amount": 4,
            "Min": null,
            "Max": null,
            "MinMaxDefine": null,
            "MinMaxApply": null,
            "AddOnAmount": null,
            "AddOnApply": null,
            "RateCurrency": "AUD",
            "DefaultExecutionType": true,
            "Tiers": [],
            "Bands": [],
            "IsActive": true
          },
          {
            "OpMic": "XASX",
            "SegMic": "XSFE",
            "BbgExchangeCode": "SFE",
            "BbgYellowKey": "Comdty",
            "BbgRootCode": "SX",
            "GmiExchangeCode": "18",
            "GmiContractType": "O",
            "GmiFuturesCode": "US",

```

```

        "ProductGroupCode": "Ag-AUD",
        "ProductCode": "US",
        "ProductType": "Option",
        "TradeType": "BlockTrade",
        "SpreadType": "Hedge",
        "RateType": "TradePriceBPS",
        "ExecutionType": "W",
        "ExecutionTypeName": "Desk",
        "Amount": 4,
        "Min": null,
        "Max": null,
        "MinMaxDefine": null,
        "MinMaxApply": null,
        "AddOnAmount": null,
        "AddOnApply": null,
        "RateCurrency": "AUD",
        "DefaultExecutionType": true,
        "Tiers": [],
        "Bands": [],
        "IsActive": true
    },
],
    "ExchangeCode": "ASX",
    "DirectBill": false
},
],
    "RateScheduleId": 13028,
    "FileName": "Test",
    "DownloadFileName": null,
    "CreatedOn": "2020-05-20T15:48:40.296Z",
    "UpdatedOn": "2024-02-29T18:07:45.246Z"
}
],
"UnsupportedProducts": [],
}

```

c. Get Updated Agreements

The Agreement Updates API allows you to retrieve agreement IDs based on API user’s party entitlements. The maximum duration allowed between the From and To date is 365 days.

Request

```
GET /AgreementApi/v2/agreement/updates?fromDate=2018-02-25T20%3A29%3A54.115Z&toDate=2018-02-27T20%3A29%3A54.115Z
```

Field	Description
FromDate	Indicates From date and time
ToDate	Indicates To date and time

Response

Field	Description
AgreementId	Unique identifier for an Agreement
AgreementType	Type of the agreement
EffectiveDate	Date on which Agreement becomes effective (this is a business date)
InitiatedOn	Initiated DateTimeStamp for the agreement in UTC
LastModified	LastModified DateTimeStamp for the agreement in UTC
State	State of the agreement
TerminationDate	Date on which agreement becomes terminated (this is a business date)

Response JSON example

```
{
  "AgreementId": 13426,
  "AgreementType": "STDC",
  "State": "Terminated",
  "InitiatedOn": "2018-02-22T21:52:28.61Z",
  "LastModified": "2018-02-26T17:05:15.673Z",
  "EffectiveDate": "2018-02-22",
  "TerminationDate": "2018-02-22"
},
{
  "AgreementId": 13431,
  "AgreementType": "STDC",
  "State": "Terminated",
  "InitiatedOn": "2018-02-26T09:41:54.76Z",
  "LastModified": "2018-02-26T09:58:06.74Z",
  "EffectiveDate": "2018-02-26",
  "TerminationDate": "2018-02-25"
},
{
  "AgreementId": 13432,
  "AgreementType": "STDC",
  "State": "Executed",
  "InitiatedOn": "2018-02-26T09:56:53.997Z",
  "LastModified": "2018-02-26T09:58:06.74Z",
  "EffectiveDate": "2018-02-26",
  "TerminationDate": null
}
```

Error Response JSON example

```
{
  "ErrorCode": "ExceededMaxDateRange",
  "ErrorMessage": "The maximum duration between 'From Date' and 'To Date' is 365 days"
}
```

d. Get Agreement Language

The Agreement Language call lets users retrieve the full agreement text based on the Agreement ID. Only a single agreement may be retrieved for each call being made.

Request

GET /AgreementApi/v2/agreement/agreementlanguage/<AgreementId>

Field	Description
AgreementId	Unique identifier for the Agreement for which the language will be retrieved

Response

Field	Description
AgreementId	Unique identifier for an Agreement
AgreementTransitionedFrom	The agreement ID from which it was cloned (may be empty if this agreement was not part of a give-up agreement transition)
AgreementTransitionedTo	The agreement ID to which it was cloned (may be empty if this agreement was not part of a give-up agreement transition)
AgreementType	Type of the agreement
BillWho	Party to which you can bill for this agreement (except Executing Broker)
BRRDVersion	Indicates which BRRDversion of FIA's standard language has been added to the agreement
CreatedOn	Date Agreement was created
CustomAgreementTemplateName	Name of the custom agreement template, if any, used on the agreement
CustomAgreementTemplateLastModifiedBy	LastModified user for the custom agreement template
CustomAgreementTemplateLastModifiedOn	LastModified DateTimeStamp for the custom agreement template
EffectiveDate	Date on which Agreement becomes effective (this is a business date)
ExecutedDate	Executed date time stamp of the agreement in UTC
FollowingAgreementId	Agreement Id to which it was amended (may be empty if this agreement was not amended)
GroupId	Unique identifier for the agreements which are related
HasAgreementTextChanges	Indicates whether or not the agreement text has been modified from the standard. Will be true for agreements with changes, false for all others

InitiatedOn	Initiated date time stamp in UTC
Jurisdiction	Agreement jurisdiction
LastModified	Last modified (touched) date time stamp in UTC
LegalLanguageParagraphs	List of paragraphs in the agreement
Paragraph10Addition	Indicates if the agreement has added paragraph 10 language for UK governing law. Will be true for agreements with changes, false for all others
Parties	List of Parties involved in the agreement
PriorAgreementId	Agreement Id from which it was amended (may be empty if this agreement is not an amendment)
State	State of the agreement
TemplateVersion	Year of the agreement template version
TerminationDate	Date on which agreement becomes terminated (this is a business date)
UploadedAgreement	Set as true if the agreement was uploaded to Docs as a PDF. false in all other cases

Response JSON example

```
{
  "CreatedOn": "2023-10-25T23:34:25.883Z",
  "ExecutedDate": "2023-10-25T23:38:08.037Z",
  "CustomerReference": "Blue-tailed Bee-eater",
  "ProfitCenter": null,
  "SuppDocCount": 0,
  "GroupId": 41518,
  "PriorAgreementId": null,
  "FollowingAgreementId": null,
  "AgreementTransitionedFrom": null,
  "AgreementTransitionedTo": null,
  "BillWho": "ClearingBroker",
  "Jurisdiction": "England and Wales",
  "Parties": [
    {
      "AgreementPartyRole": "ClearingBroker",
      "PartyFirms": [],
      "IsConformed": true,
      "PartyId": 100113,
      "Name": "Gringotts Wizarding Bank",
      "ParentId": 0,
      "InstitutionId": 0,
      "IsActive": true,
      "LegalEntityIdentifier": null,
      "CreatedOn": "2018-07-25T20:23:05.166Z",
      "UpdatedOn": "2024-01-10T15:42:26.41Z"
    },
    {
      "AgreementPartyRole": "ExecutionBroker",
      "PartyFirms": [],
      "IsConformed": false,
      "PartyId": 100383,
    }
  ]
}
```

```

    "Name": "TestpartyNonConforming",
    "ParentId": 0,
    "InstitutionId": 0,
    "IsActive": true,
    "LegalEntityIdentifier": null,
    "CreatedOn": "2019-01-30T16:47:02.876Z",
    "UpdatedOn": "2023-10-24T09:56:54.966Z"
  },
  {
    "AgreementPartyRole": "Trader",
    "PartyFirms": [],
    "IsConformed": false,
    "PartyId": 100478,
    "Name": "Blue-tailed Bee-eater",
    "ParentId": 100476,
    "InstitutionId": 100476,
    "IsActive": true,
    "LegalEntityIdentifier": null,
    "CreatedOn": "2020-03-10T16:39:17.256Z",
    "UpdatedOn": "2020-03-10T16:39:17.353Z"
  }
],
"HasAgreementTextChanges": true,
"Paragraph10Addition": false,
"BrrdVersion": null,
"AgreementId": 41518,
"AgreementType": "STDT",
"TemplateVersion": "2017",
"State": "Executed",
"InitiatedOn": "2023-10-25T23:34:25.997Z",
"LastModified": "2023-10-25T23:38:08.033Z",
"EffectiveDate": "2023-10-25",
"TerminationDate": null
"UploadedAgreement": false,
"LegalLanguageParagraphs": [
  {
    "Order": 1,
    "HasAgreementTextChanges": false,
    "Text": "All transactions executed orally, in writing or through an
electronic order facility or cleared hereunder shall be subject to
applicable laws, governmental, regulatory, self-regulatory, exchange or
clearing house rules, regulations, interpretations, protocols and the
customs and usages of the exchange or clearing house on which they are
executed and cleared, as in force from time to time ("Applicable Law"). All
disputes relating to transactions executed or cleared under this Agreement
shall be governed by and settled pursuant to Applicable Law and shall be
subject to the jurisdiction of the exchange (and, if applicable, its
clearing house) upon which the dispute arises. The parties to this
Agreement shall perform their respective obligations and exercise their
respective rights under this Agreement (including, but not limited to,
rejecting a Customer order, calling a Customer for margin or providing any
notice specified herein) using commercially reasonable judgement, in a
commercially reasonable manner under the circumstances, and consistent with
Applicable Law."
  }
],

```

```

{
  "Order": 2,
  "HasAgreementTextChanges": true,
  "Text": "Customer authorizes Executing Broker to execute orders for
Customer as transmitted orally, in writing or through an electronic order
facility by Trader/Additional Trader to Executing Broker, or, as permitted
by Applicable Law, directly to an exchange. Executing Broker reserves the
right to reject an order that Trader/Additional Trader may transmit to
Executing Broker for execution and shall promptly notify Trader/Additional
Trader of any such rejection. Clearing Broker may, upon prior notice to
Executing Broker and Trader/Additional Trader, place limits or conditions on
the positions it will accept for give-up for Customer's account. In the
event this Agreement is executed on behalf of Customer by Trader, Trader
hereby represents and warrants that Trader is expressly authorized by
Customer to enter into this Agreement on Customer's behalf, and that this
authorization will remain effective unless Customer or Trader provides prior
written notice to Executing Broker and Clearing Broker."
}
]
}

```

Error Response JSON example

```

{
  "ErrorCode": "InvalidAgreementId",
  "ErrorMessage": "Agreement Id is not valid"
}

```

```

{
  "ErrorCode": "AgreementNotFound",
  "ErrorMessage": "Agreement not found"
}

```

Response JSON example for a DELETED agreement

```

{
  "CreatedOn": "2023-05-16T18:55:49.293Z",
  "ExecutedDate": null,
  "CustomerReference": null,
  "ProfitCenter": null,
  "SuppDocCount": 0,
  "GroupId": 0,
  "PriorAgreementId": null,
  "FollowingAgreementId": null,
  "AgreementTransitionedFrom": null,
  "AgreementTransitionedTo": null,
  "BillWho": null,
  "Jurisdiction": null,
  "Parties": null,
  "ClearingAccounts": null,
  "RateSchedules": null,
}

```

```

"UnsupportedProducts": null,
"HasAgreementTextChanges": false,
"Paragraph10Addition": false,
"BrrdVersion": null,
"AgreementId": 40915,
"AgreementType": null,
"TemplateVersion": null,
"State": "Deleted",
"InitiatedOn": null,
"LastModified": "2023-08-28T21:00:30.38Z",
"EffectiveDate": null,
"TerminationDate": null
}

```

e. Put Agreement Comments

This agreement based API call retrieves comments made on agreements. Comments can be retrieved for up to 1,000 agreements at a time. An optional time-date parameter allows for the request to pull comments made only during a specific time range.

Request

PUT /AgreementApi/v2/agreementComments

Field	Description
AgreementIds	Up to 1,000 unique Agreements for which comments will be retrieved
FromDate	Optional From date and time. If ToDate is provided, FromDate must be as well.
ToDate	Optional To date and time. If FromDate is provided, ToDate must be as well

Response

Field	Description
AgreementId	Unique identifier for an Agreement
AgreementsNotFound	List of agreement IDs included in the request that were not found and will not be returned in the response
AgreementType	Type of the agreement
BillWho	Party to which you can bill for this agreement (except Executing Broker)
CreatedOn	Date Agreement was created
Comments	All comments on the agreement for the requested time period

EffectiveDate	Date on which Agreement becomes effective (this is a business date)
ExecutedDate	Executed date time stamp of the agreement in UTC
FollowingAgreementId	Agreement Id to which it was amended (may be empty if this agreement was not amended)
GroupId	Unique identifier for the agreements which are related
InitiatedOn	Initiated date time stamp in UTC
Jurisdiction	Agreement jurisdiction
LastModified	Last modified (touched) date time stamp in UTC
PriorAgreementId	Agreement Id from which it was amended (may be empty if this agreement is not an amendment)
State	State of the agreement
TemplateVersion	Year of the agreement template version
TerminationDate	Date on which agreement becomes terminated (this is a business date)

Response JSON example

```
{
  "AgreementsNotFound": "631",
  "AgreementsAndComments": [
    {
      "CreatedOn": "2024-09-03T19:49:06.91Z",
      "ExecutedDate": "2024-09-03T20:06:46.33Z",
      "GroupId": 52429,
      "PriorAgreementId": 52429,
      "FollowingAgreementId": 62979,
      "BillWho": "ClearingBroker",
      "Jurisdiction": "The Jungle",
      "AgreementId": 62978,
      "AgreementType": "STDT",
      "TemplateVersion": "2017",
      "State": "Executed",
      "InitiatedOn": "2024-09-03T19:49:06.257Z",
      "LastModified": "2025-01-13T22:10:31.847Z",
      "EffectiveDate": "2024-09-03",
      "TerminationDate": null
    },
    {
      "CreatedOn": "2025-01-13T22:10:20.167Z",
      "CreatedBy": {
        "UserName": "Harry Potter",
        "UserEmail": "potter@fia.tech",
        "UserParty": "Gringotts Wizarding Bank - EU division"
      },
      "Workstream": "Clearing Accounts",
      "ToParty": null,
      "CommentType": "Comment",
      "CommentDetails": "Clearing Account added to agreement to reflect Virtucon's new fund"
    }
  ]
}
```

```

    },
  ],
}
]
}

```

f. Put Agreement History

This agreement based API call retrieves agreement history records. The history records can be retrieved for up to 1,000 agreements at a time. An optional time-date parameter allows for the request to pull records of changes made during a specific time period.

Request

PUT /AgreementApi/v2/agreementHistory

Field	Description
AgreementIds	Up to 1,000 unique Agreements for which history will be retrieved
FromDate	Optional From date and time. If ToDate is provided, FromDate must be as well.
ToDate	Optional To date and time. If FromDate is provided, ToDate must be as well

Response

Field	Description
AgreementId	Unique identifier for an Agreement
AgreementsNotFound	List of agreement IDs included in the request that were not found and will not be returned in the response
AgreementType	Type of the agreement
BillWho	Party to which you can bill for this agreement (except Executing Broker)
CreatedOn	Date Agreement was created
EffectiveDate	Date on which Agreement becomes effective (this is a business date)
ExecutedDate	Executed date time stamp of the agreement in UTC
FollowingAgreementId	Agreement Id to which it was amended (may be empty if this agreement was not amended)
GroupId	Unique identifier for the agreements which are related
HistoryRecords	Actions taken on the agreement as recorded in the agreement history
InitiatedOn	Initiated date time stamp in UTC
Jurisdiction	Agreement jurisdiction
LastModified	Last modified (touched) date time stamp in UTC

PriorAgreementId	Agreement Id from which it was amended (may be empty if this agreement is not an amendment)
State	State of the agreement
TemplateVersion	Year of the agreement template version
TerminationDate	Date on which agreement becomes terminated (this is a business date)

Response JSON example

```
{
  "AgreementsNotFound": "396",
  "AgreementsAndHistoryRecords": [
    {
      "CreatedOn": "2024-12-12T16:13:58.28Z",
      "ExecutedDate": "2024-12-12T17:21:00.983Z",
      "GroupId": 52423,
      "PriorAgreementId": 52482,
      "FollowingAgreementId": 63322,
      "BillWho": "ClearingBroker",
      "Jurisdiction": "The Jungle",
      "AgreementId": 63321,
      "AgreementType": "STDC",
      "TemplateVersion": "2017",
      "State": "Executed",
      "InitiatedOn": "2024-12-12T16:13:58.067Z",
      "LastModified": "2024-12-12T17:21:00.977Z",
      "EffectiveDate": "2024-12-13",
      "TerminationDate": null
      "HistoryRecords": [
        {
          "CreatedOn": "2024-12-12T17:21:11.36Z",
          "CreatedBy": {
            "UserName": "DefaultPartyLimits",
            "UserEmail": "",
            "UserParty": "Gringotts Wizarding Bank"
          },
          "ActionDetails": "Section 2 Notice issued"
        },
        {
          "CreatedOn": "2024-12-12T17:21:03.37Z",
          "CreatedBy": {
            "UserName": "Harry Potter",
            "UserEmail": "potter@fia.tech",
            "UserParty": "Gringotts Wizarding Bank"
          },
          "ActionDetails": "Agreement change - Agreement executed"
        },
        {
          "CreatedOn": "2024-12-12T17:21:01.453Z",
          "CreatedBy": {
            "UserName": "Harry Potter",
            "UserEmail": "potter@fia.tech",
            "UserParty": "Gringotts Wizarding Bank"
          }
        }
      ]
    }
  ]
}
```

```

    },
    "ActionDetails": "Agreement Effective date set to 13-Dec-2024"
  },
  {
    "CreatedOn": "2024-12-12T17:18:46.26Z",
    "CreatedBy": {
      "UserName": "Spike Marlin",
      "UserEmail": "smarlin@fia.tech.com",
      "UserParty": "Warbucks Munitions Division"
    },
    "ActionDetails": "Agreement change - Agreement executed by Warbucks Munitions Division (Customer)"
  },
  {
    "CreatedOn": "2024-12-12T17:11:56.737Z",
    "CreatedBy": {
      "UserName": "Tycho Nestoris",
      "UserEmail": "tnestoris@fia.tech.com",
      "UserParty": "Iron Bank of Braavos"
    },
    "ActionDetails": "Agreement change - Agreement executed by Iron Bank of Braavos (Executing Broker)"
  },
  {
    "CreatedOn": "2024-12-12T16:13:59.847Z",
    "CreatedBy": {
      "UserName": "Tycho Nestoris",
      "UserEmail": "tnestoris@fia.tech.com",
      "UserParty": "Iron Bank of Braavos"
    },
    "ActionDetails": "Agreement change - Rate schedule Standard Rates added"
  }
],
}
]
}

```

g. Put Agreement Workstream Status

This agreement based API call retrieves the current workstream status for up to 1,000 agreements at a time. It includes information on when a workstream was approved, and by whom.

Request

PUT /AgreementApi/v2/agreementWorkStreamStatus

Field	Description
AgreementIds	Up to 1,000 unique Agreements for which history will be retrieved

Response

Field	Description
AgreementId	Unique identifier for an Agreement
AgreementsNotFound	List of agreement IDs included in the request that were not found and will not be returned in the response
AgreementType	Type of the agreement
BillWho	Party Role being billed based on the agreement (except Executing Broker)
CreatedOn	Date Agreement was created
EffectiveDate	Date on which Agreement becomes effective (this is a business date)
ExecutedDate	Executed date time stamp of the agreement in UTC
FollowingAgreementId	Agreement Id to which it was amended (may be empty if this agreement was not amended)
GroupId	Unique identifier for the agreements which are related
HistoryRecords	Actions taken on the agreement as recorded in the agreement history
InitiatedOn	Initiated date time stamp in UTC
Jurisdiction	Agreement jurisdiction
LastModified	Last modified (touched) date time stamp in UTC
Parties	List of Parties involved in the agreement
PriorAgreementId	Agreement Id from which it was amended (may be empty if this agreement is not an amendment)
State	State of the agreement
TemplateVersion	Year of the agreement template version
TerminationDate	Date on which agreement becomes terminated (this is a business date)
WorkstreamApprovals	Information on which workstream approvals have been set and by which users

Response JSON example

```
{
  "AgreementsNotFound": "23",
  "AgreementWorkstreamStatus": [
    {
      "CreatedOn": "2025-01-09T22:49:52.493Z",
      "ExecutedDate": null,
      "GroupId": 52451,
      "PriorAgreementId": 52451,
      "FollowingAgreementId": null,
      "BillWho": "ClearingBroker",
      "Jurisdiction": "The Jungle",
      "AgreementId": 63351,
      "AgreementType": "STDT",
      "TemplateVersion": "N/A",
      "State": "Execution",
    }
  ]
}
```

```

"InitiatedOn": "2025-01-09T22:49:52.397Z",
"LastModified": "2025-01-09T22:50:13.84Z",
"EffectiveDate": null,
"TerminationDate": null
"Parties": [
  {
    "PartyId": 100112,
    "Name": "Iron Bank of Braavos",
    "IsConformed": false,
    "IsActive": false,
    "AgreementPartyRole": "Clearing Broker",
    "WorkstreamApprovals": [
      {
        "Type": "Execution",
        "Approved": false,
        "ApprovedOn": null,
        "ApprovedBy": null
      }
    ]
  },
  {
    "PartyId": 100114,
    "Name": "Tellson's Bank (London)",
    "IsConformed": true,
    "IsActive": false,
    "AgreementPartyRole": "Executing Broker",
    "WorkstreamApprovals": [
      {
        "Type": "Legal Language",
        "Approved": true,
        "ApprovedOn": "2024-05-09T15:58:57.51Z",
        "ApprovedBy": "ewiklund_sim"
      },
      {
        "Type": "Rate Schedule",
        "Approved": true,
        "ApprovedOn": "2025-01-09T22:50:12.123Z",
        "ApprovedBy": "jlorry"
      },
      {
        "Type": "Clearing Accounts",
        "Approved": true,
        "ApprovedOn": "2024-05-09T15:58:58.546Z",
        "ApprovedBy": "ewiklund_sim"
      },
      {
        "Type": "Execution",
        "Approved": false,
        "ApprovedOn": null,
        "ApprovedBy": null
      }
    ]
  },
  {
    "PartyId": 121627,

```

```

    "Name": "Warbucks Shipping Company",
    "IsConformed": true,
    "IsActive": false,
    "AgreementPartyRole": "Trader",
    "WorkstreamApprovals": [
      {
        "Type": "Legal Language",
        "Approved": true,
        "ApprovedOn": "2024-05-09T15:51:22.28Z",
        "ApprovedBy": "smarlin"
      },
      {
        "Type": "Rate Schedule",
        "Approved": true,
        "ApprovedOn": "2025-01-09T22:50:13.123Z",
        "ApprovedBy": "smarlin"
      },
      {
        "Type": "Clearing Accounts",
        "Approved": true,
        "ApprovedOn": "2024-05-09T15:54:18.753Z",
        "ApprovedBy": "smarlin"
      },
      {
        "Type": "Execution",
        "Approved": false,
        "ApprovedOn": null,
        "ApprovedBy": null
      }
    ]
  },
],
}
]
}

```

10. Clearing Accounts

These calls described in this section can only be performed by the Clearing Broker to a Give-Up Agreement. Note that the below validations apply:

- Delete clearing account is only available for agreements in Composition and Execution states
- Account and agreement uniqueness must be respected
- Account dates must fall within agreement dates

a. Add clearing account

The Add clearing account API adds a clearing account to an agreement

Request

POST /DocsApi/agreement/clearingAccount

Field	Description
AgreementId	Unique identifier for an Agreement
ClearingAccount	ClearingAccount object to add to the agreement

Request JSON example

```
{
  "AgreementId": 12345,
  "ClearingAccounts": [
    {
      "ClearingAccountType": "CustomerClearingAccount",
      "AccountId": "AAGGVV",
      "Description": "SomeClearingAccount",
      "EffectiveDate": "2019-01-10",
      "ExpirationDate": "2019-02-11"
    }
  ]
}
```

Response

HTTP Code 200

Returns 200 status code if the clearing account was successfully added on the agreement.

Error Response JSON example

```
{
  "ErrorCode": "InvalidAgreementId",
  "ErrorMessage": "Agreement Id is not valid"
}
```

```
{
  "ErrorCode": "InvalidClearingAccountId",
  "ErrorMessage": "Clearing Account Id is not valid"
}
```

```
{
  "ErrorCode": "InvalidClearingAccountId",
  "ErrorMessage": "Clearing Account Id is not valid"
}
```

b. Add multiple clearing accounts

The Add multiple clearing accounts API adds multiple clearing accounts to an agreement.

Request

POST /DocsApi/agreement/clearingAccounts

Field	Description
AgreementId	Unique identifier for an Agreement
ClearingAccounts	List of ClearingAccount object to add to the agreement

```
{
  "AgreementId": 12345,
  "ClearingAccounts": [
    {
      "ClearingAccountType": "CustomerClearingAccount",
      "AccountId": "AAGGVV",
      "Description": "SomeClearingAccount",
      "EffectiveDate": "2019-01-10",
      "ExpirationDate": "2019-02-11"
    },
    {
      "ClearingAccountType": "FcmProcessingReference",
      "AccountId": "BBGGVV",
      "Description": "SomeClearingAccount2",
      "EffectiveDate": "2019-01-010",
      "ExpirationDate": "2019-02-11"
    }
  ]
}
```

Response

HTTP Code 200

Returns 200 status code if the range of clearing accounts was successfully added on the agreement.

Error Response JSON example

```
{
  "ErrorCode": "InvalidAgreementId",
  "ErrorMessage": "Agreement Id is not valid"
}
```

```
{
  "ErrorCode": "InvalidClearingAccountId",
  "ErrorMessage": "Clearing Account Id is not valid"
}
```

```
{
  "ErrorCode": "InvalidClearingAccountId",
  "ErrorMessage": "Clearing Account Id is not valid"
}
```

```
}
```

c. Delete clearing account

The delete clearing account API deletes a clearing account from an agreement.

Request

DELETE

/DocsApi/agreement/clearingAccount?agreementId=123456&clearingAccountId=AABVCFD

Field	Description
AgreementId	Unique identifier for an Agreement
ClearingAccountId	The account Id of the clearing account

Response

HTTP Code 200

Returns 200 status code if the clearing account was successfully deleted from the agreement.

Error Response JSON example

```
{
  "ErrorCode": "InvalidAgreementId",
  "ErrorMessage": "Agreement Id is not valid"
}
```

```
{
  "ErrorCode": "InvalidClearingAccountId",
  "ErrorMessage": "Clearing Account Id is not valid"
}
```

d. Expire clearing account

The expire clearing account API adds an expiration date to an account on an agreement. The date passed as the ExpirationDate gets added to the account on the agreement.

Request

PUT

**/DocsApi/agreement/expireClearingAccount?agreementId=123456&clearingAccountId=AA
BVCFD&expirationDate=2018-02-27**

Field	Description
AgreementId	Unique identifier for an Agreement
ClearingAccountId	The account Id of the clearing account
ExpirationDate	The date on which to expire the clearing account (only the date component)

Response

HTTP Code 200

Returns 200 status code if the clearing account was successfully expired on the specified date.

Error Response JSON example

```
{
  "ErrorCode": "InvalidAgreementId",
  "ErrorMessage": "Agreement Id is not valid"
}
```

```
{
  "ErrorCode": "InvalidClearingAccountId",
  "ErrorMessage": "Clearing Account Id is not valid"
}
```

```
{
  "ErrorCode": "InvalidBusinessDate",
  "ErrorMessage": " Effective Dates, Expiration Dates and Termination Date
cannot have time component "
}
```

11. Agreement Risk

The Agreement Risk calls in this section are only available to users subscribing to the Risk API. For additional information contact FIA Tech Client Services at docs.support@fia-tech.com or +1 202-772-3000.

a. Get Agreement Risk Details

The Agreement Risk Details API allows you to request agreement risk details based on API user's party entitlements.

Request

```
GET /DocsApi/agreementRisk/details?agreementId=13423
```

Field	Description
AgreementId	Unique identifier for the Agreement to the details for.

Response

Field	Description
AgreementId	Unique identifier for an Agreement
RiskLastModified	DateTimeStamp in UTC when risk was last updated on agreement
RiskStatus	Indicates the Risk Status
Limits	List of Risk Limits
Conditions	List of Risk Conditions
ProductExclusions	List of Risk Product Exclusions

Response JSON example

```
{
  "AgreementId": 13423,
  "RiskLastModified": "2019-04-08T16:16:16.611Z",
  "RiskStatus": "RestrictedDueToTradeByTrade",
  "Limits": [
    {
      "Type": "DailyInitialMargin",
      "Currency": "USD",
      "Amount": 10000,
      "LimitEffectiveDate": "2019-04-08T16:16:16.611Z",
      "CreatedOn": "2019-04-08T16:16:16.611Z",
      "Version": 1,
      "State": "Draft"
    },
    {
      "Type": "DailyMaxContracts",
      "Currency": "",
      "Amount": 500,
      "LimitEffectiveDate": "2019-04-08T16:16:16.611Z",
      "CreatedOn": "2019-04-08T16:16:16.611Z",
      "Version": 1,
      "State": "Approved"
    }
  ],
  "Conditions": [
    {
      "Type": "TradeByTradeReview",
      "Details": "",
      "CreatedOn": "2019-04-08T16:16:16.611Z"
    },
    {
      "Type": "Custom",
      "Details": "Some Detail",
      "CreatedOn": "2019-04-08T16:16:16.611Z"
    }
  ]
}
```

```

"ProductExclusions": [
  {
    "Region": "North America",
    "ProductName": "Cboe Bitcoin (USD) Futures",
    "EffectiveDate": "2019-04-08T16:16:16.611Z",
    "ExchangeCode": "CFE",
    "ProductGroupCode": "EqIdx-USD-Crypto",
    "ProductCode": "XBT",
    "ProductType": "Future"
  },
  {
    "Region": "North America",
    "ProductName": "Bitcoin Futures",
    "EffectiveDate": "2019-04-08T16:16:16.611Z",
    "ExchangeCode": "CME",
    "ProductGroupCode": "EqIdx-US-CP",
    "ProductCode": "BTC",
    "ProductType": "Future"
  },
  {
    "Region": "North America",
    "ProductName": "Bitcoin Futures",
    "EffectiveDate": "2019-04-08T16:16:16.611Z",
    "ExchangeCode": "CME",
    "ProductGroupCode": "EqIdx-US",
    "ProductCode": "BTC",
    "ProductType": "Future"
  }
]

```

Error Response JSON example

```

{
  "ErrorCode": "InvalidAgreementId",
  "ErrorMessage": "Agreement Id is not valid"
}

```

```

{
  "ErrorCode": "AgreementNotFound",
  "ErrorMessage": "Agreement not found"
}

```

b. Get Agreement Risk Updates

The Get Agreement Risk Updates API allows you to retrieve agreement IDs on which risk components were updated between specified dates. The maximum duration allowed between the From and To date is 1095 days (~ 3 years).

Request

```
GET /DocsApi/agreementRisk/updates?fromDate=2018-02-25T20%3A29%3A54.115Z&toDate=2018-02-27T20%3A29%3A54.115Z
```

Field	Description
FromDate	Indicates From date and time
ToDate	Indicates To date and time

Response

Field	Description
AgreementId	Unique identifier for an Agreement
RiskLastModified	DateTimeStamp in UTC when risk was last updated on agreement

Response JSON example

```
{
  "AgreementId": 13426,
  "RiskLastModified": "2018-02-22T21:52:28.61Z",
},
{
  "AgreementId": 13440,
  "RiskLastModified": "2018-05-11T20:42:30.50Z",
},
{
  "AgreementId": 13459,
  "RiskLastModified": "2018-07-17T10:18:22.20Z",
}
```

Error Response JSON example

```
{
  "ErrorCode": "ExceededMaxDateRange",
  "ErrorMessage": "The maximum duration between 'From Date' and 'To Date' is 1095 days"
}
```

c. Set Limits

The Set limits API adds a risk limit to an agreement. Only Clearing Brokers can update limits or conditions on agreements.

Request

```
POST /DocsApi/agreementRisk/limits
```

Field	Description
AgreementId	Unique identifier for an Agreement
Type	Risk Limit Type Enum
Currency	Currency code of limit amount
Amount	Limit amount

```
{
  "AgreementId": 12345,
  "Type": "DailyInitialMargin",
  "Currency": "USD",
  "Amount": 50000,
}
```

Response

HTTP Code 200

Returns 200 status code if the risk limit was successfully added on the agreement.

Error Response JSON example

```
{
  "ErrorCode": "InvalidAgreementTypeStateForRisk",
  "ErrorMessage": "Agreement Type or State is not valid for this operation"
}
```

```
{
  "ErrorCode": "MissingAgreementRiskLimitType",
  "ErrorMessage": "Agreement Risk Limit Type is required"
}
```

```
{
  "ErrorCode": "InvalidAgreementRiskLimitType",
  "ErrorMessage": "Agreement Risk Limit Type is not valid. Allowed values are 'DailyInitialMargin' and 'DailyMaxContracts'"
}
```

```
{
  "ErrorCode": "MissingAgreementRiskLimitCurrency",
  "ErrorMessage": "Currency is required if Agreement Risk Limit Type is Daily Initial Margin"
}
```

```
{
  "ErrorCode": "InvalidAgreementRiskLimitCurrency",
  "ErrorMessage": "Currency is not valid"
}
```

```
{
  "ErrorCode": "AgreementRiskLimitCurrencyNotRequired",
  "ErrorMessage": "Currency is not required if Agreement Risk Limit Type
is Daily Max Contracts"
}
```

```
{
  "ErrorCode": "MissingAgreementRiskLimitAmount",
  "ErrorMessage": "Agreement Risk Limit Amount is required"
}
```

```
{
  "ErrorCode": "InvalidAgreementRiskLimitAmount",
  "ErrorMessage": "Agreement Risk Limit Amount is not valid"
}
```

```
{
  "ErrorCode": "AgreementRiskLimitAlreadyExists",
  "ErrorMessage": "This Agreement Risk Limit already exists"
}
```

d. Delete Limits

The Delete limits API deletes a risk limit from an agreement. Only Clearing Brokers can update limits or conditions on agreements.

Passing type as 'All' will delete all the risk limits from the agreement.

Request

```
DELETE /DocsApi/agreementRisk/limits?agreementId=13243&type=DailyInitialMargin
```

Field	Description
AgreementId	Unique identifier for an Agreement
Type	Risk Limit Type Enum or All

Response

```
HTTP Code 200
```

Returns 200 status code if the risk limit was successfully deleted from the agreement.

Error Response JSON example

```
{
  "ErrorCode": "InvalidAgreementTypeStateForRisk",
  "ErrorMessage": "Agreement Type or State is not valid for this
operation"
}
```

```
{
  "ErrorCode": "MissingAgreementRiskLimitType",
  "ErrorMessage": "Agreement Risk Limit Type is required"
}
```

```
{
  "ErrorCode": "InvalidAgreementRiskLimitType",
  "ErrorMessage": "Agreement Risk Limit Type is not valid. Allowed values
are 'DailyInitialMargin' and 'DailyMaxContracts'"
}
```

```
{
  "ErrorCode": "AgreementRiskLimitNotFound",
  "ErrorMessage": "Agreement Risk Limit not found"
}
```

e. Set Conditions

The Set conditions API adds a risk condition to an agreement. Only Clearing Brokers can update limits or conditions on agreements.

Passing type as 'Custom' also requires the 'Details' to be passed.

Request

POST /DocsApi/agreementRisk/conditions

Field	Description
AgreementId	Unique identifier for an Agreement
Type	Risk Condition Type Enum
Details	Condition details – Only required for custom conditions

```
{
  "AgreementId": 12345,
  "Type": "TradeByTradeReview",
  "Details": "",
}
```

Response

HTTP Code 200

Returns 200 status code if the risk condition was successfully added on the agreement.

Error Response JSON example

```
{
  "ErrorCode": "InvalidAgreementTypeStateForRisk",
  "ErrorMessage": "Agreement Type or State is not valid for this
operation"
}
```

```
{
  "ErrorCode": "MissingAgreementRiskConditionType",
  "ErrorMessage": "Agreement Risk Condition Type is required"
}
```

```
{
  "ErrorCode": "InvalidAgreementRiskConditionType",
  "ErrorMessage": "Agreement Risk Condition Type is not valid. Allowed
values are 'AgreementSuspended', 'TradeByTradeReview' and 'Custom'"
}
```

```
{
  "ErrorCode": "MissingAgreementRiskConditionDetails",
  "ErrorMessage": "Agreement Risk Condition Details is required"
}
```

```
{
  "ErrorCode": "AgreementRiskConditionDetailsNotRequired",
  "ErrorMessage": "Agreement Risk Condition Details is not required if
Type is not Custom"
}
```

```
{
  "ErrorCode": "AgreementRiskConditionAlreadyExists",
  "ErrorMessage": "This Agreement Risk Condition already exists"
}
```

f. Delete Conditions

The Delete conditions API deletes a risk condition from an agreement.

Passing type as 'All' will delete all the risk conditions from the agreement. Passing type as 'Custom' also requires the 'Details' to be passed to match with an existing custom condition.

Request

DELETE

/DocsApi/agreementRisk/conditions?agreementId=13243&type=TradeByTradeReview

OR

/DocsApi/agreementRisk/conditions?agreementId=13243&type=Custom&details=someDetail

Field	Description
AgreementId	Unique identifier for an Agreement
Type	Risk Condition Type Enum or All
Details	Condition details – Only required for custom condition

Response

HTTP Code 200

Returns 200 status code if the risk condition was successfully deleted from the agreement.

Error Response JSON example

```
{
  "ErrorCode": "InvalidAgreementTypeStateForRisk",
  "ErrorMessage": "Agreement Type or State is not valid for this operation"
}
```

```
{
  "ErrorCode": "MissingAgreementRiskConditionType",
  "ErrorMessage": "Agreement Risk Condition Type is required"
}
```

```
{
  "ErrorCode": "InvalidAgreementRiskConditionType",
  "ErrorMessage": "Agreement Risk Condition Type is not valid. Allowed values are 'AgreementSuspended', 'TradeByTradeReview' and 'Custom'"
}
```

```
{
  "ErrorCode": "MissingAgreementRiskConditionDetails",
  "ErrorMessage": "Agreement Risk Condition Details is required"
}
```

```
{
  "ErrorCode": "AgreementRiskConditionDetailsNotRequired",
}
```

```
"ErrorMessage": "Agreement Risk Condition Details is not required if
Type is not Custom"
}
```

```
{
  "ErrorCode": "AgreementRiskConditionNotFound",
  "ErrorMessage": "Agreement Risk Condition not found"
}
```

g. Set Product Exclusions

The Set product exclusions API adds a risk Product Exclusion to an agreement. Only Clearing Brokers can update limits or conditions on agreements.

Request

POST /DocsApi/agreementRisk/productExclusions

Field	Description
AgreementId	Unique identifier for an Agreement
ExchangeCode	Exchange code
ProductGroupCode	Product group code
ProductCode	Product code
ProductType	Product type

```
{
  "AgreementId": 12345,
  "ExchangeCode": "CFE",
  "ProductGroupCode": "EqIdx-USD-Crypto",
  "ProductCode": "CFE",
  "ProductType": "Future"
}
```

Response

HTTP Code 200

Returns 200 status code if the risk product exclusions was successfully added on the agreement.

Error Response JSON example

```
{
  "ErrorCode": "InvalidAgreementTypeStateForRisk",
  "ErrorMessage": "Agreement Type or State is not valid for this
operation"
}
```

```
{
  "ErrorCode": "ProductNotFound",
  "ErrorMessage": "Product not found"
}
```

```
{
  "ErrorCode": "AgreementRiskProductExclusionAlreadyExists",
  "ErrorMessage": "This Agreement Risk Product Exclusion already exists"
}
```

h. Delete Product Exclusions

The Delete product exclusions API deletes a product exclusion from an agreement. Only Clearing Brokers can update limits or conditions on agreements.

Request

DELETE /DocsApi/agreementRisk/productExclusions?deleteAllProductExclusions=true

Field	Description
AgreementId	Unique identifier for an Agreement
ExchangeCode	Exchange code
ProductGroupCode	Product group code
ProductCode	Product code
ProductType	Product type
DeleteAllProductExclusions	Optional – Pass true to delete all product exclusions

```
{
  "AgreementId": 12345,
  "ExchangeCode": "CFE",
  "ProductGroupCode": "EqIdx-USD-Crypto",
  "ProductCode": "CFE",
  "ProductType": "Future"
}
```

Response

HTTP Code 200

Returns 200 status code if the risk product exclusions was successfully deleted from the agreement.

Error Response JSON example

```
{
  "ErrorCode": "InvalidAgreementTypeStateForRisk",
  "ErrorMessage": "Agreement Type or State is not valid for this operation"
}
```

```
}

```

```
{
  "ErrorCode": "ProductNotFound",
  "ErrorMessage": "Product not found"
}
```

```
{
  "ErrorCode": "AgreementRiskProductExclusionNotFound",
  "ErrorMessage": "Agreement Risk Product Exclusion not found"
}
```

12. Rate Schedule

a. Upload Rate Schedule

The Rate schedule upload API allows you to upload a rate schedule to a party's rate schedule library.

Request

POST /DocsApi/rateschedule/upload?partyId=257

Field	Description
Exchanges	List of Exchanges where this Rate Schedule is valid
FileName	The name of the Rate Schedule to be displayed in the Rate Schedule Library

Note: ExecutionTypeName is optional in Upload Rate Schedule request, but if provided it must be a valid ExecutionTypeName matching the ExecutionType in request.

Request JSON example

```
{
  "Exchanges": [
    {
      "Rates": [
        {
          "ProductGroupCode": "EqSSO-EUR",
          "ProductCode": "All",
          "ProductType": "Option",
          "Rank": null,
          "TradeType": null,
          "SpreadType": null,
          "RateType": "PerContract",
          "ExecutionType": "Y",
          "ExecutionTypeName": "Electronic",

```

```

        "Amount": 0.7,
        "Min": null,
        "Max": null,
        "MinMaxDefine": null,
        "MinMaxApply": null,
        "AddOnAmount": null,
        "AddOnApply": null,
        "RateCurrency": "EUR",
        "DefaultExecutionType": true,
        "Tiers": [ ],
        "Bands": [ ],
        "IsActive": false
    },
    {
        "ProductGroupCode": "EqSSF-EUR",
        "ProductCode": "All",
        "ProductType": "Future",
        "Rank": null,
        "TradeType": null,
        "SpreadType": null,
        "RateType": "PerContract",
        "ExecutionType": "Y",
        "ExecutionTypeName": "Electronic",
        "Amount": 0,
        "Min": null,
        "Max": null,
        "MinMaxDefine": null,
        "MinMaxApply": null,
        "AddOnAmount": null,
        "AddOnApply": null,
        "RateCurrency": "EUR",
        "DefaultExecutionType": true,
        "Tiers": [ ],
        "Bands": [ ],
        "IsActive": true
    }
],
"ExchangeCode": "EUREX",
"DirectBill": false
},
{
    "Rates": [
        {
            "ProductGroupCode": "EqSSF-CAD",
            "ProductCode": "All",
            "ProductType": "Future",
            "Rank": null,
            "TradeType": null,
            "SpreadType": null,
            "RateType": "PerContract",
            "ExecutionType": "Y",
            "ExecutionTypeName": "Electronic",
            "Amount": 0,
            "Min": null,
            "Max": null,

```

```

        "MinMaxDefine": null,
        "MinMaxApply": null,
        "AddOnAmount": null,
        "AddOnApply": null,
        "RateCurrency": "CAD",
        "DefaultExecutionType": true,
        "Tiers": [ ],
        "Bands": [ ],
        "IsActive": false
    },
    ],
    "ExchangeCode": "IFLO",
    "DirectBill": false
},
"FileName": "MONDAY_v10.xlsm"
}

```

Response

HTTP Code 200

Returns 200 status code if the rate schedule was successfully uploaded.

Error Response JSON example

```

{
  "ErrorCode": "MissingRateScheduleName",
  "ErrorMessage": "Rate Schedule name is not provided"
}

```

```

{
  "ErrorCode": "InvalidRateScheduleName",
  "ErrorMessage": "Rate Schedule does not exist"
}

```

```

{
  "ErrorCode": "NotFound",
  "ErrorMessage": "Not found"
}

```

```

{
  "ErrorCode": "MissingRateScheduleExchanges",
  "ErrorMessage": "Rate Schedule exchanges are not provided"
}

```

```

{
  "ErrorCode": "DuplicateExchangeCode",
  "ErrorMessage": "Exchange code is duplicated"
}

```

```
{  
  "ErrorCode": "InvalidRank",  
  "ErrorMessage": "Rank should be sequential and unique"  
}
```

```
{  
  "ErrorCode": "InvalidAmount",  
  "ErrorMessage": "Amount is not accepted if Rate Type is Vol Disc or  
Band"  
}
```

```
{  
  "ErrorCode": "MissingAmount",  
  "ErrorMessage": "Amount is required if Rate Type is not Vol Disc and  
Band"  
}
```

```
{  
  "ErrorCode": "MissingDefaultExecutionType",  
  "ErrorMessage": "Default Execution Type is required"  
}
```

```
{  
  "ErrorCode": "MissingExecutionType",  
  "ErrorMessage": "ExecutionType is required"  
}
```

```
{  
  "ErrorCode": "InvalidExecutionType",  
  "ErrorMessage": "ExecutionType is not valid"  
}
```

```
{  
  "ErrorCode": "InvalidExecutionTypeName",  
  "ErrorMessage": "ExecutionTypeName is not matching the provided  
ExecutionType"  
}
```

```
{  
  "ErrorCode": "InvalidRateType",  
  "ErrorMessage": "Rate Type is required"  
}
```

```
{
  "ErrorCode": "InvalidProductDefinitionCodes",
  "ErrorMessage": "Error while getting Product from"
}
```

b. Get Updated Rate Schedules

The Update Rate Schedule API allows you to retrieve a list of filenames based on the updated date of the rate schedules.

Request

```
GET /RateScheduleApi/v2/rateschedule/updates?fromDate=2018-04-10T20%3A29%3A54.115Z&toDate=2018-05-26T20%3A29%3A54.115Z&partyId=257
```

Field	Description
FromDate	Indicates From date and time
ToDate	Indicates To date and time
PartyId	Optional – Unique identifier for the party the rate schedule to be retrieved should be associated with. If this is not passed, rate schedule updates for all the permitted parties will be returned.

Response

Field	Description
FileName	The name of the Rate Schedule as displayed in the Rate Schedule Library
CreatedOn	Indicates created on datetimestamp
UpdatedOn	Indicates updated on datetimestamp

Response JSON example

```
{
  "FileName": "MONDAY_v10.xlsm",
  "CreatedOn": "2018-03-09T12:33:17.36Z",
  "UpdatedOn": "2018-03-10T12:33:17.36Z",
}
```

c. Download Rate Schedule

The Rate schedule download API allows you to download a rate schedule from a party's rate schedule library. This is the standard API download and will return the rate schedule with product group level rate rows as they exist on the rate schedule. Similar to the [Expanded Agreement Details API](#) call, an [Expanded](#)

[Rate Schedule](#) call exists as well that will break down group level product rows to include product level detail and any cross-reference symbology to which the firm subscribes.

Request

```
GET
/RateScheduleApi/v2/rateschedule/download?partyId=257&fileName=MONDAY_v10.xlsm
```

Field	Description
PartyId	Unique identifier for the party the rate schedule to be downloaded should be associated with
FileName	The name of the Rate Schedule in the Rate Schedule Library

Response

Field	Description
Exchanges	List of Exchanges where this Rate Schedule is valid
ExecutingAccounts	List of Executing Accounts available on this rate schedule
FileName	The downloaded rate schedule's filename

Response JSON example

```
{
  "Exchanges": [
    {
      "Rates": [
        {
          "ProductGroupCode": "Ag-Grain&Oilseed",
          "ProductCode": "All",
          "ProductType": "Mixed",
          "TradeType": null,
          "SpreadType": null,
          "RateType": "PerContract",
          "ExecutionType": "Y",
          "ExecutionTypeName": "Electronic",
          "Amount": 0.7,
          "Min": null,
          "Max": null,
          "MinMaxDefine": null,
          "MinMaxApply": null,
          "AddOnAmount": null,
          "AddOnApply": null,
          "RateCurrency": "USD",
          "DefaultExecutionType": true,
          "Tiers": [ ],
          "Bands": [ ],
          "IsActive": false
        }
      ]
    }
  ]
}
```

```

        "ProductGroupCode": "Ag-Grain&Oilseed",
        "ProductCode": "All",
        "ProductType": "Mixed",
        "TradeType": null,
        "SpreadType": null,
        "RateType": "PerContract",
        "ExecutionType": "W",
        "ExecutionTypeName": "Electronic",
        "Amount": 0.5,
        "Min": null,
        "Max": null,
        "MinMaxDefine": null,
        "MinMaxApply": null,
        "AddOnAmount": null,
        "AddOnApply": null,
        "RateCurrency": "USD",
        "DefaultExecutionType": false,
        "Tiers": [ ],
        "Bands": [ ],
        "IsActive": true
    },
    ],
    "ExchangeCode": "CME",
    "DirectBill": false
},
],
"ExecutingAccounts": null,
"FileName": "MONDAY_v10.xlsm",
"CreatedOn": "2018-03-09T12:33:17.36Z ",
"UpdatedOn": null
}

```

Error Response JSON example

```

{
  "ErrorCode": "RateScheduleNotFound",
  "ErrorMessage": "Rate Schedule not found"
}

```

d. Download Expanded Rate Schedule

The Expanded Rate Schedule call includes all the information of the [standard Rate Schedule](#) call, plus the group level rate lines broken out into individual products and enriched with cross-reference product information.

The call can be used in two manners. Either the symbology parameters can all be left unspecified, in which case the response will return to the user all symbologies to which the firm is subscribed, or a specific set of symbologies can be specified. To include a specific symbology, the value for the requested data should be set to “true”, otherwise it should be set as “false”.

Regardless of which symbologies are specified, exchange product information along with segment and operating mic will always be included for all products.

For information on how to download the rate schedules associated with agreements containing this information, please see the [Expanded Agreement Details](#) section in this document.

Request

GET

```
/RateScheduleApi/v2/rateschedule/download/expanded?partyId=257&fileName=MONDAY_v10.xlsm
```

Field	Description
PartyId	Unique identifier for the party the rate schedule to be downloaded should be associated with
FileName	The name of the Rate Schedule in the Rate Schedule Library
Bbg	Indication to include bbg symbologies in response
Gmi	Indication to include gmi symbologies in response

Response

Field	Description
Exchanges	List of Exchanges where this Rate Schedule is valid
ExecutingAccounts	List of Executing Accounts available on this rate schedule
FileName	The downloaded rate schedule's filename

Response JSON example

```
{
  "Exchanges": [
    {
      "Rates": [
        {
          "OpMic": "XCME",
          "SegMic": "XCME",
          "BbgExchangeCode": "CME",
          "BbgYellowKey": "Comdty",
          "BbgRootCode": "MDS",
          "GmiExchangeCode": "02",
          "GmiContractType": null,
          "GmiFuturesCode": "ZZ",
          "ProductGroupCode": "Ag-Grain&Oilseed",
          "ProductCode": "CPO",
          "ProductType": "Future",
          "TradeType": null,
          "SpreadType": null,
          "RateType": "PerContract",
          "ExecutionType": "Y",
        }
      ]
    }
  ]
}
```

```

        "ExecutionTypeName": "Electronic",
        "Amount": 0.7,
        "Min": null,
        "Max": null,
        "MinMaxDefine": null,
        "MinMaxApply": null,
        "AddOnAmount": null,
        "AddOnApply": null,
        "RateCurrency": "USD",
        "DefaultExecutionType": true,
        "Tiers": [ ],
        "Bands": [ ],
        "IsActive": true
    },
    {
        "OpMic": "XCME",
        "SegMic": "XCME",
        "BbgExchangeCode": "CME",
        "BbgYellowKey": "Comdty",
        "BbgRootCode": "VPC",
        "GmiExchangeCode": "02",
        "GmiContractType": null,
        "GmiFuturesCode": "AO",
        "ProductGroupCode": "Ag-Grain&Oilseed",
        "ProductCode": "CPV",
        "ProductType": "Future",
        "TradeType": null,
        "SpreadType": null,
        "RateType": "PerContract",
        "ExecutionType": "Y",
        "ExecutionTypeName": "Electronic",
        "Amount": 0.7,
        "Min": null,
        "Max": null,
        "MinMaxDefine": null,
        "MinMaxApply": null,
        "AddOnAmount": null,
        "AddOnApply": null,
        "RateCurrency": "USD",
        "DefaultExecutionType": true,
        "Tiers": [ ],
        "Bands": [ ],
        "IsActive": true
    }
],
    "ExchangeCode": "CME",
    "DirectBill": false
},
    ],
    "ExecutingAccounts": null,
    "FileName": "MONDAY_v10.xlsm",
    "CreatedOn": "2018-03-09T12:33:17.36Z ",
    "UpdatedOn": null
}

```

Error Response JSON example

```
{  
  "ErrorCode": "RateScheduleNotFound",  
  "ErrorMessage": "Rate Schedule not found"  
}
```

e. Delete Rate Schedule

The Rate schedule delete API allows deleting a rate schedule from a party's rate schedule library.

Request

```
DELETE /DocsApi/rateschedule/delete?partyId=257&fileName=MONDAY_v10.xlsm
```

Field	Description
PartyId	Unique identifier for the party the rate schedule to be deleted should be associated with
FileName	Filename of the Rate Schedule to be deleted

Response

HTTP Code 200

Returns 200 status code if the rate schedule was successfully deleted.

Error Response JSON example

```
{  
  "ErrorCode": "RateScheduleNotFound",  
  "ErrorMessage": "Rate Schedule not found"  
}
```

13. Reference Data

a. Get Parties

The Get parties API retrieves a list of parties based on the comma-separated list of PartyIds passed and/or based on the From and/or To date range passed which match the last modified timestamp for the party. If any part of the request is not passed or cannot be parsed, then the response will include all the entitled parties.

Request

```
PUT /DocsApi/referenceDataApi/v2/parties
```

```
{
  "PartyIds": [
    100123,
    100125
  ],
  "FromDate": "2018-06-24T03:23:01.852Z",
  "ToDate": "2018-08-24T03:23:01.852Z"
}
```

Response

Field	Description
Code	Alphanumeric unique code for the party
CreatedOn	Date when the party was created
LegalEntityIdentifier	LEI for the party
InstitutionId	InstitutionId of the party. If the party is not a sub-party, this will be identical to the ParentId
IsActive	Indicates whether the Party is active or inactive
Name	Name of the party
ParentId	Parent PartyId of the party. If the party is not a sub-party, this will be identical to the InstitutionId
PartyId	Unique identifier for the party
UpdatedOn	Date when the party was last updated

Response JSON example

```
{
  "PartyId": 100123,
  "Name": "Very Big Corp. of America",
  "ParentId": 0,
  "InstitutionId": 0,
  "LegalEntityIdentifier": 123,
  "IsActive": false,
  "CreatedOn": "2018-07-25T20:23:05.167Z",
  "UpdatedOn": null
},
{
  "PartyId": 100125,
  "Name": "Warbucks Industries",
  "ParentId": 0,
  "InstitutionId": 0,
  "LegalEntityIdentifier": 456,
  "IsActive": false,
  "CreatedOn": "2018-07-25T20:23:05.167Z",
  "UpdatedOn": null
}
```

b. Get Exchanges

The Get Exchanges API retrieves a list of exchanges based on the exchange code passed and/or based on the From and/or To date range passed which match the last modified datetime for the exchange. If any part of the request is not passed or cannot be parsed, then the response will include all the exchanges.

Request

PUT /DocsApi/referenceDataApi/v2/exchanges

```
{
  "ExchangeCodes": [
    "AFET",
    "IFLL"
  ]
}
```

Response

Field	Description
Code	Alphanumeric unique code for the exchange
Name	Name of the exchange
CreatedOn	Date when the exchange was created
UpdatedOn	Date when the exchange was last updated

Response JSON example

```
{
  "Code": "AFET",
  "Name": "AGRICULTURAL FUTURES EXCHANGE OFTHAILAND",
  "CreatedOn": "2007-11-16T21:44:00.417Z",
  "UpdatedOn": "2007-11-16T21:44:00.417Z"
},
{
  "Code": "IFLL",
  "Name": "ICE Futures Europe Financial Products",
  "CreatedOn": "2014-09-16T14:43:15.353Z",
  "UpdatedOn": "2014-09-16T14:43:15.353Z"
}
```

c. Get Exchange Firms

The Get Exchange Firms API retrieves a list of exchange firms based on the PartyId passed and/or based on the From and/or To date range passed which match the last modified datetime for the party. If any part of the request is not passed or cannot be parsed, then the response will include all the exchange firms.

Request

PUT /DocsApi/referenceDataApi/v2/exchangeFirms

```
{
  "PartyIds": [
    195
  ]
}
```

Response

Field	Description
PartyFirms	List of PartyFirms
PartyId	Unique identifier for the party

Response JSON example

```
{
  "PartyFirms": [
    {
      "FirmCode": "CbotFirm1",
      "ExchangeCode": "CBOT",
      "PrimaryFlag": "1",
      "CreatedOn": "2018-07-19T00:00:00Z",
      "UpdatedOn": null
    },
    {
      "FirmCode": "CbotFirm2",
      "ExchangeCode": "CBOT",
      "PrimaryFlag": "1",
      "CreatedOn": "2018-07-18T00:00:00Z",
      "UpdatedOn": null
    }
  ],
  "PartyId": 195
}
```

d. Get Exchange Product Groups

The Get Exchange Product Groups API retrieves a list of exchange product groups based on the Exchange Code passed and/or based on the From and/or To date range passed which match the last modified datetime for the Exchange. If any part of the request is not passed or cannot be parsed, then the response will include all the exchange product groups.

Request

PUT /DocsApi/referenceDataApi/v2/exchangeProductGroups

```
{
  "ExchangeCodes": [
    "AFET"
  ]
}
```

Response

Field	Description
ProductGroups	List of Product Groups for the exchange
ExchangeCode	The unique alphanumeric code for the Exchange

Response JSON example

```
{
  "ProductGroups": [
    {
      "ProductGroupCode": "Ag-THB",
      "Products": [
        {
          "ProductCode": "RSS3",
          "ProductName": "Natural Rubber Ribbed Smoked Sheet No.3
Futures",
          "ProductType": "Future",
          "CreatedOn": "2009-02-06T19:02:59.44Z",
          "UpdatedOn": "2009-02-06T19:02:59.44Z"
        },
        {
          "ProductCode": "STR20",
          "ProductName": "Block Rubber STR20 Futures",
          "ProductType": "Future",
          "CreatedOn": "2009-02-06T19:02:59.443Z",
          "UpdatedOn": "2009-02-06T19:02:59.443Z"
        }
      ]
    },
    {
      "ProductGroupCode": "ALL-F",
      "Products": [
        {
          "ProductCode": "All",
          "ProductName": "ALL FUTURES",
          "ProductType": "Future",
          "CreatedOn": "2012-02-10T21:00:11Z",
          "UpdatedOn": "2012-02-10T21:00:11Z"
        }
      ]
    }
  ],
  {
    "ProductGroupCode": "All",
```

```

        "Products": [
            {
                "ProductCode": "All",
                "ProductName": "All",
                "ProductType": "Future",
                "CreatedOn": "2012-02-10T21:00:12Z",
                "UpdatedOn": "2012-02-10T21:00:12Z"
            }
        ]
    },
    "ExchangeCode": "AFET"
}

```

e. Get Enumerations

The Get Enumerations API allows to retrieve all the valid enums used in Docs Api.

Request

GET /DocsApi/referenceDataApi/v2/enumerations

Response

Field	Description
AgreementPartyRole	List of AgreementPartyRole enum values
AgreementState	List of AgreementState enum values
AgreementType	List of AgreementType enum values
BandValueType	List of BandValueType enum values
ClearingAccountType	List of ClearingAccountType enum values
ExecutionType	List of ExecutionType enum values
MinMaxDefine	List of MinMaxDefine enum values
PartyRole	List of PartyRole enum values
Period	List of Period enum values
ProductType	List of ProductType enum values
RateApply	List of RateApply enum values
RateApplyTo	List of RateApplyTo enum values
RateType	List of RateType enum values
SpreadType	List of SpreadType enum values
StartType	List of StartType enum values
TradeType	List of TradeType enum values

Response JSON example

```

{
    "AgreementState": [
        "Pending",
    ]
}

```

```

    "Draft",
    "Composition",
    "Execution",
    "Executed",
    "Terminated",
    "Stale",
    "Deleted",
    "PendingInitiation",
    "InitiationRejected",
    "CompositionLocked"
  ],
  "AgreementType": [
    "STDC",
    "STDT",
    "STDO",
    "EFPT",
    "EFPC",
    "GUSA",
    "CLSA",
    "LMECCB",
    "LMECXM",
    "LMECXMCB",
    "LMECEAC",
    "LMECCAC",
    "LMECEACCAC",
    "LMECEACCB",
    "LMECEACCACCB",
    "LMETCB",
    "LMETXM",
    "LMETXMCB",
    "LMETEAC",
    "LMETCAC",
    "LMETEACCAC",
    "LMETEACCB",
    "LMETEACCACCB",
    "LMEC",
    "LMET"
  ],
  "AgreementPartyRole": [
    "AdditionalTrader",
    "Trader",
    "ExecutionBroker",
    "ClearingBroker",
    "Customer",
    "ExecutingAdministrativeClearer",
    "ClearingAdministrativeClearer",
    "CarryingBroker",
    "OrderPassingBroker",
    "LMEClearingMember",
    "LMEExecutingMember",
    "AdditionalTrader",
    "Any"
  ],
  "ClearingAccountType": [
    "CustomerClearingAccount",

```

```

    "FcmProcessingAccount"
  ],
  "ProductType": [
    "COMBO",
    "Forwrđ",
    "FUT",
    "Future",
    "Mixed",
    "OFF",
    "OOC",
    "OOF",
    "OPT",
    "Option",
    "PHY",
    "Spot",
    "Swap",
    "Swaps"
  ],
  "TradeType": [
    "BlockTrade",
    "ExchangeForPhysical",
    "ExchangeForRisk",
    "ExchangeForSwap",
    "BasisTrade",
    "PerAllocation"
  ],
  "SpreadType": [
    "Spread",
    "PitSpread",
    "Roll",
    "Hedge",
    "Fly",
    "Straddle",
    "Strangle",
    "Switch",
    "TAS",
    "Condor"
  ],
  "RateType": [
    "PerContract",
    "PerTrade",
    "VolDisc",
    "PercentPremium",
    "PercentNotional",
    "BPS",
    "MWH",
    "BTU",
    "MBTU",
    "MMBTU",
    "TONNES",
    "Barrels",
    "MetricTons",
    "Vega",
    "Band"
  ],
],

```

```

"MinMaxDefine": [
  "Amount",
  "PercentPremium",
  "PercentNotional"
],
"RateApply": [
  "PerContract",
  "PerTrade"
],
"RateApplyTo": [
  "UseOnlyOne",
  "UseAllThatApply"
],
"Period": [
  "Daily",
  "Monthly"
],
"StartType": [
  "Amount",
  "Number",
  "PremiumPrice",
  "NotionalPrice",
  "NumberOfContracts",
  "NumberOfTrades"
],
"BandValueType": [
  "BPS",
  "Notional",
  "NumberOfContracts",
  "Premium",
  "TradePrice"
],
"ExecutionType": [
  '{ "ExecutionType": "Y", "ExecutionTypeName": "Electronic" }',
  '{ "ExecutionType": "Z", "ExecutionTypeName": "ClientPit" }',
  '{ "ExecutionType": "W", "ExecutionTypeName": "Desk" }',
  '{ "ExecutionType": "X", "ExecutionTypeName": "DeskPit" }',
  '{ "ExecutionType": "A", "ExecutionTypeName": "PhoneSimple" }',
  '{ "ExecutionType": "B", "ExecutionTypeName": "PhoneComplex" }',
  '{ "ExecutionType": "LC", "ExecutionTypeName": "FcmProvidedScreen" }',
  '{ "ExecutionType": "C", "ExecutionTypeName":
  'VendorProvidedPlatformBilledByExecutingBroker' }',
  '{ "ExecutionType": "LD", "ExecutionTypeName":
  'OtherProvidedScreen' }',
  '{ "ExecutionType": "D", "ExecutionTypeName":
  'OtherIncludingOtherProvidedScreen' }',
  '{ "ExecutionType": "E", "ExecutionTypeName":
  'ClientProvidedPlatformControlledByFcm' }',
  '{ "ExecutionType": "H", "ExecutionTypeName":
  'PremiumAlgorithmicTradingProviderBilledByExecutingBroker' }',
  '{ "ExecutionType": "F", "ExecutionTypeName":
  'ClientProvidedPlatformDirectToExchange' }',
  '{ "ExecutionType": "LG", "ExecutionTypeName": "FcmApiOrFix" }',
  '{ "ExecutionType": "G", "ExecutionTypeName":
  'SponsoredAccessViaExchangeApiOrFixProvidedByExecutingBroker' }',

```

```

    "{ 'ExecutionType': 'LH', 'ExecutionTypeName': 'AlgoEngine' }",
    "{ 'ExecutionType': 'J', 'ExecutionTypeName': 'PriceAtExecution' }",
    "{ 'ExecutionType': 'K', 'ExecutionTypeName': 'K' }",
    "{ 'ExecutionType': 'M', 'ExecutionTypeName': 'M' }",
    "{ 'ExecutionType': 'N', 'ExecutionTypeName': 'N' }",
    "{ 'ExecutionType': 'P', 'ExecutionTypeName': 'P' }",
    "{ 'ExecutionType': 'Q', 'ExecutionTypeName': 'Q' }",
    "{ 'ExecutionType': 'R', 'ExecutionTypeName': 'R' }",
    "{ 'ExecutionType': 'S', 'ExecutionTypeName': 'S' }",
    "{ 'ExecutionType': 'T', 'ExecutionTypeName': 'T' }",
    "{ 'ExecutionType': 'U', 'ExecutionTypeName': 'U' }",
    "{ 'ExecutionType': 'V', 'ExecutionTypeName': 'V' }",
    "{ 'ExecutionType': '2', 'ExecutionTypeName': '2' }",
    "{ 'ExecutionType': '3', 'ExecutionTypeName': '3' }",
    "{ 'ExecutionType': '4', 'ExecutionTypeName': '4' }",
    "{ 'ExecutionType': '5', 'ExecutionTypeName': '5' }",
    "{ 'ExecutionType': '6', 'ExecutionTypeName': '6' }",
    "{ 'ExecutionType': '7', 'ExecutionTypeName': '7' }",
    "{ 'ExecutionType': '8', 'ExecutionTypeName': '8' }",
    "{ 'ExecutionType': '9', 'ExecutionTypeName': '9' }"
  ],
  "PartyRole": [
    "Trader",
    "ExecutionBroker",
    "ClearingBroker",
    "Customer",
    "Any"
  ]
}

```

14. User

a. Generate New Api Key

Generates a new Api Key for user. The previous key becomes invalid after new key is successfully generated.

Request

GET /DocsApi/user/generateNewApiKey

Response

Field	Description
ApiKey	New Api Key
ExpiresOn	Expiration date of newly generated key



Response JSON example

```
{  
  "ApiKey": "e6ca9011096e4644a6a6c5dd655c9d4c",  
  "ExpiresOn": "01-Oct-2020"  
}
```