

MATS API apps Terms Of Reference & Conformance Test Guide

EPEX SPOT
12.09.2025

Content

1.	Introduction	4
1.1	Audience	4
1.2	Purpose	4
1.3	Obligation to cooperate	4
1.4	Context	4
1.5	Document revision history	5
2.	MATS API apps Conformance Test	6
2.1	Goal	6
2.2	Who must go through a Conformance Test	6
2.3	Process overview	7
2.4	Mandatory declaration of API app (applicable for all customers)	7
2.5	Pre-requisites	7
2.6	Register for a Conformance Test	8
2.7	Conformance Test environment	8
2.8	Protocol / Operational steps (conf test day)	8
3.	Terms Of Reference (ToR) requirements	11
3.1	Requests	11

1. Introduction

1.1 Audience

This MATS REST API Client Application Design Guide is meant for:

- developers coding client applications to communicate with the MATS REST API,
- any other profile like project managers who would like to understand which rules have to be respected by API apps to pass the MATS API conformance test and be authorized to operate in production.

1.2 Purpose

The purpose of this document is to describe the expected behavior of API applications using the MATS API Public message interface. It is a set of rules and requirements which need to be adhered and applied to using the MATS API.

These rules will ensure the proper functioning of interactions between client apps on exchange member side or non-market participant side (e.g. Data Vendors) and MATS on the side of EPEX SPOT.

1.3 Obligation to cooperate

The rules mentioned in this document should be respected by application's developers. EPEX SPOT reserves the right to block access to the MATS API server and disconnect customers' API users from the EPEX SPOT system (MATS API server and MATS server) any time if the application of the member endangers the stability of the Trading System.

1.4 Context

Since 11 November 2024, the *Terms of Reference* were included in the MATS REST API Client Application Design Guide, as a specific section within the document:

- 1) **The Push¹ functionality** Please refer to the MATS Push Design Guide) **is recommended to optimize your API application, but not mandatory**. Therefore, you can secure your ETS app migration to MATS by first implementing the REST API, and optimize it later on with the push part, to avoid loops, especially by subscribing to orders and market results notifications, before or after the go live.
- 2) **Implementation requirements:**
 - a. Your app should start, right after getting connected, by doing an **initialization/starting sequence**, sending all the requests it needs to potentially get configuration and market data (user and auction configuration, your orders, published market results not yet retrieved, etc.). Please refer to the Sample Test Client implementation, illustrating this point.

¹ Please refer to the MATS Push Client Applications Design Guide

b. Then refresh data when required:

- i. if you do not implement the Push, implement “loops” to periodically refresh data (*).
- ii. If your app implements the Push, it must react to the received notifications by sending the corresponding request to refresh your data on the scope indicated in the notifications. Please refer to the Push Design guide for further details on authorized ad hoc / on demand requests.

(*) : when not using the Push and implementing your requests “loops”, please make the **duration** (in seconds) **between 2 identical requests configurable**: e.g. to get orders for a specific type on specific scope, to get results for a given auction session, etc.

This stand-alone document precises and illustrates the main points of these ToR :

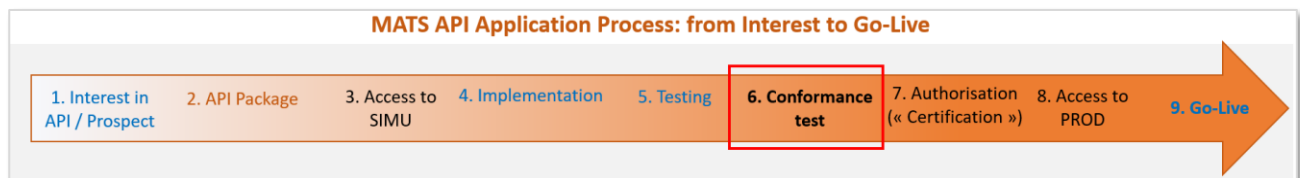
- Time interval (number of seconds) to respect between 2 identical requests, per requests “categories”.
- Details about published market results not yet retrieved, focusing on the link between auction-session requests to get result status and the sending of actual market results retrieval (which do not contain results’ status like in the ETS API)
- Details the Conformance test “protocol/operational steps”, which is the series of steps to be followed by MATS API apps on their conformance test day.

1.5 Document revision history

Evolution	Date	Version	Author	Comments
Creation	07.04.2025	1.0	EPEX SPOT API Customer Services	Stand-alone version of Terms Of Reference, developing the generic aspects : • Defines the interval in seconds to respect between 2 identical requests, • Illustrating diagrams.
Update	05.05.2025	1.1	EPEX SPOT API Customer Services	Precisions about intervals: • per user • examples about identical and non-identical requests.
Update	25.07.2025	1.2	EPEX SPOT API Customer Services	Intervals table updated for Trading Limits and FTS orders.
Update	14.08.2025	1.3	EPEX SPOT API Customer Services	Include dedicated chapter of API app declaration which is mandatory Explicitly mention the scenarios which are disregarded as not a ToR breach for identical GET requests Minor rephrasing of statements for better clarity

2. MATS API apps Conformance Test

The API Conformance Test Phase is a **mandatory milestone** in a MATS API application life cycle:



Any new API application must go through this test.

2.1 Goal

The goal of MATS API apps Conformance test goal is to **accompany API clients best towards Production**. Its principles remain the same as for the ETS API:

- Conformance to our **Terms Of Reference** (“ToR”, see below)
 - ToR are slightly simplified (e.g. login policy), but principles remain the same: **chasing “abusive” automatic/periodic patterns** (e.g. asking results when they are for sure not yet available). Ad hoc/on demand requests are always allowed (e.g. to refresh data in a screen).
- Accompany you best, not block you from going live:
 - conformance tests can be passed with a caveat, giving you a delay (usually a few weeks) to solve the issue, depending on the impact of the issue on the MATS API & system, without preventing you from operating your app in production.
 - applications with major issues, e.g. flooding MATS with tons of unnecessary requests breaching our ToR, are not authorized to go live.

The Conformance test is designed to **verify the interactions between a MATS API application and the MATS API**. It focuses on ensuring that the app adheres MATS ToR. However, it **does not evaluate the internal logic or functionality of the app itself**.

2.2 Who must go through a Conformance Test

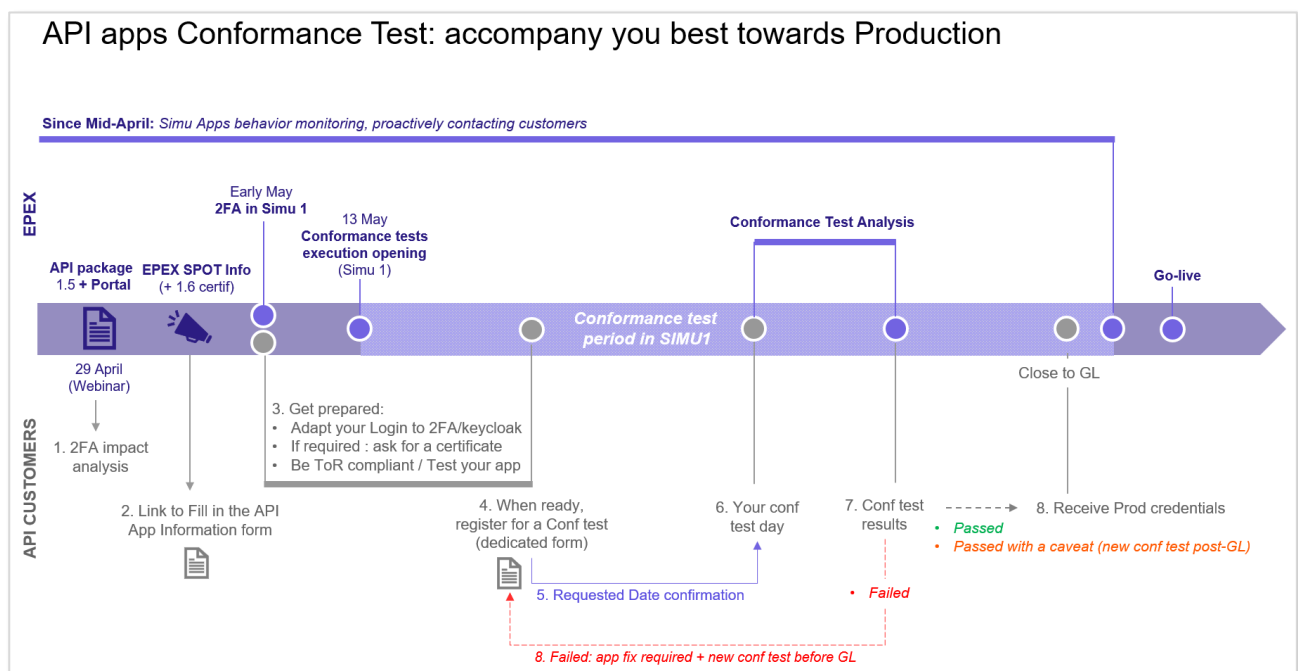
- 1) **Every MATS API client implementing In-house application** must go through a conformance test before going live:
 - Members and Non Market Participants (Market Data Clients called “Data Vendors”) using **in-house** API application
 - **If you use an ISV app: the conformance test [for this ISV app](#) must only be done by your ISV.**

- If you are using both an ISV app and an own/custom app, your own app must go through a conformance test.

2) Multi-exchanges situations

Clients using their app with several exchanges only need to go through the conformance test for one exchange (if Epex + another exchange : do the test via Epex).

2.3 Process overview



2.4 Mandatory declaration of API app (applicable for all customers)

- **Fill in MATS API app & Client Info Information Form:** at the beginning of your app implementation activities, please inform us the details about your API application so we can reference it internally and support you best.
 - If you are using in-house API application (developed by internally or using external service provider)
 - If you are using ISV application

2.5 Pre-requisites

- **Conformance test Testing Pre-requisites: your application must be already completely tested** on your own before you apply for a conformance test. Please

contact Market Operations to perform any required action (e.g. market orders entry, running the auction) needed to complete your internal test phase.

- **Is 2FA mandatory to go through the conformance test?**
 - Only the 2FA log in to the MATS system (connect) is mandatory.
 - You can implement the rest (log out, reset and change password, get user info to know your password expiry) and test it autonomously later on.
 - Please note that you must be able to reset your user password via Keycloak GUI manually for the MATS go live, otherwise you will not be able to connect.

2.6 Register for a Conformance Test

When ready (please see the testing Pre-requisites above and Terms Of reference below), please contact the Exchange to register for a conformance test **via the dedicated MATS API Conformance Test Registration Form**.

2.7 Conformance Test environment

Conformance test only take place on the **SIMU1** environment.

2.8 Protocol / Operational steps (conf test day)

Scope: depending on the functional scope of your API application there may be steps that need to be skipped since irrelevant.

Please find below the steps that need to be followed for the conformance test day.

Conformance Test Steps	Read only applications	Read Write Applications (able to place orders)
1. Preparation	• Environment: setup your application to connect to the environment indicated in the above section 2.7. At the agreed date: • Log in when the Orderbook (Auction Session) of the agreed auction(s) for the test is OPEN (e.g. in the morning of the test day) • Ensure the API user used during the test is not used by any other of your MATS API application at the same time. General Guideline : behave like you would do in production: • Do not log in and out constantly like you would do when testing an application, just operate the application like you would do in real conditions, • Submit as much as possible production-like orders for a more realistic test.	
	In case your RO app retrieves private market results/reports or own orders, you must submit orders.	In case your app retrieves private market results/reports or own orders, you must submit orders.
	Place orders for the target auction(s) with another user:	Place orders for the target auction(s):

	a) via the MATS Web Client, b) or via another Read Write API app (custom/in-house or ISV software) .	a) with your API app b) and any other app which is part of your production setup (MATS Web Client, custom/in-house or ISV software)
	If required, the Market operations team will also place orders in the market in order to create trades for you (i.e. to ensure that the orders you placed are executed when running the auction).	
2. Login Policy	- Do not log in and out constantly like you would do when testing an application, just operate the application like you would do in real conditions, - As explained in the Login policy section, Login (connect) and logout (disconnect) requests are not to be initiated for single requests. A user session usually shall span a larger amount of requests. - Please refer to the related section for more details. Log in (connect) with 2FA is mandatory: - The first authentication factor is your user/password - The 2nd factor is your client certificate. (please refer to the MATS API Certificate document in the API package)	
3. Requests sending policy	- Your app automated requests loops (automatic periodic requests) should respect the interval and logic defined in the below ToR. - In particular: <u>for apps retrieving results/reports</u>, loops must be done at Auction Sessions level, getting as a response each auction session status and results status. Then, when the results status indicates new results (preliminary or final) are available, the relevant public and private results/reports should be sent. - This means that your app must not send periodic market results/reports requests. - If using the push, your app should react to a received notification by sending one or several requests to retrieve the corresponding data. - Ensure your API app can retrieve public and private market results/reports (whatever is relevant to your business) as well "on demand" (as per <i>ToR</i>)	
4. Log out & close your app ?	- Behave like you would do in production: - If your app has a GUI and you would log out after having completed this sequence, just log out and close your application. - If your app is a robot, for instant aiming at retrieving data like market results when relevant, just let it run as in production. - There is no obligation to log out or close your application.	
5. Conformance Test analysis & results	- Market Operations team will start ASAP checking the logs of your API to assess if your API app respects the <i>Terms of References</i> - Once your requests are analyzed the Market Operations team will contact you to provide you with the conformance test results: a) Passed : i) your app is authorized to be operated in production, b) Passed with a caveat : i) you may be authorized to operate your application in production, ii) but you have a certain period to implement some adjustments and potentially go through a new conformance test (when relevant the corrected behavior can also be checked directly in production). c) Failed : i) your app is not authorized to be operated in production,	

	ii) explanations are given on what needs to be adjusted before attempting a new conformance test.
6. Next Steps	• In case you passed the conformance test you will receive your production credentials when the production environment is available and open to customers. • You need a dedicated MATS PROD Certificate for the Production environment. Please provide the Market operations team with the .CSR (Certificate Signing Request) • Please refer to the Certificate document in the API package for more details.

3. Terms Of Reference (ToR) requirements

3.1 Requests

All these rules only apply to automated periodic requests (called loops).

Please note that **ad hoc/on demand requests are always allowed**. Some examples

- **For API apps with a GUI:**
 - *To populate a screen with data (initial data retrieval/display),*
 - *On demand of an end user (e.g. potential forced manual screen refresh, e.g. forcing public market results retrieval).*
- **Order submission process:**
 - *If the app send GET request during the order submission then it is likelihood that due to immediate POST/DELETE requests, the application send GET requests before and after POST/DELETE requests to know the order book status, the status of orders etc. This is not regarded as repetitive pattern.*

3.1.1 Requests sending policy

3.1.1.1 Interval between requests

MATS API apps must respect a minimum time interval (number of seconds) **between 2 identical and periodic** GET requests (exact same api request with same input/query parameters, e.g. retrieving *auction session information* for the same *auction session Id*). This is applicable per user.

Note:

In case an app must restart (e.g. due to a disconnection or any recovery reason), it may happen that the intervals mentioned above are not respected. However, this will not be considered a ToR breach, as it occurs occasionally and is not regarded as a repetitive pattern.

Request Types	GET Request category (*)	Minimum interval to respect between 2 identical auto/periodic requests (per user)
GET	Back Office Configuration	60 seconds
	Auction Configuration	
	FX Rates	
	Auction session	30 seconds (Auction session & orders: use the push to get real time info)
	Orders (including FTS)	
	Trading Limits	
	Logs	

	Public Market Results Private Results Reports	- No periodic loops on results requests. - The loop must be done at auction sessions level, and results requests sent only when detecting a <i>result status</i> change for a specific Auction Session. (e.g. <i>UNAVAILABLE</i> → <i>PRELIMINARY</i>)
POST	Authentication: Connect, Disconnect	N/A (see login/logout policy)
	Authentication: Password management	N/A
POST / DELETE	Orders	N/A

(*) please refer to the MATS Swagger online portal to get the list of requests corresponding to a “category”

Examples of non-identical requests:

Example 1:

- These 2 Get requests are considered as not identical/different:
 - [/api/auction-sessions/32/linear-orders](#)
 - [/api/auction-sessions/33/linear-orders](#)
- Meaning each of these Get requests can be sent automatically by the same user with at least 30 seconds between each other:
 - [/api/auction-sessions/32/linear-orders](#) with its own cycle/loop (minimum 30 sec between 2)
 - [/api/auction-sessions/33/linear-orders](#) with its own cycle/loop (minimum 30 sec between 2)

Example 2: with query parameters:

- These 2 Get requests are considered as not identical:
 - [/api/auction-sessions?delivery-date=2025-05-02](#)
 - [/api/auction-sessions?delivery-date=2025-05-03](#)

Example 3: these 3 Get linear orders and block orders all belong to the same category, but are different :

- [/api/auction-sessions/123/linear-orders](#)
- [/api/auction-sessions/234/linear-orders](#)
- [/api/auction-sessions/123/block-orders](#)

3.1.1.2 Login policy

Login (connect) and/or logout (disconnect) requests are not to be initiated for single requests. A user session usually shall span a larger amount of requests.

Do NOT	DO instead
To perform 2 operations, do not insert a login and logout between each operation: - log in perform operation 1 (e.g. get market results for auction session A) - log out - log in again perform operation 2 (e.g. get market results for auction session B) - log out	To perform 2 operations, do: - log in perform operation 1 perform operation 2 - log out

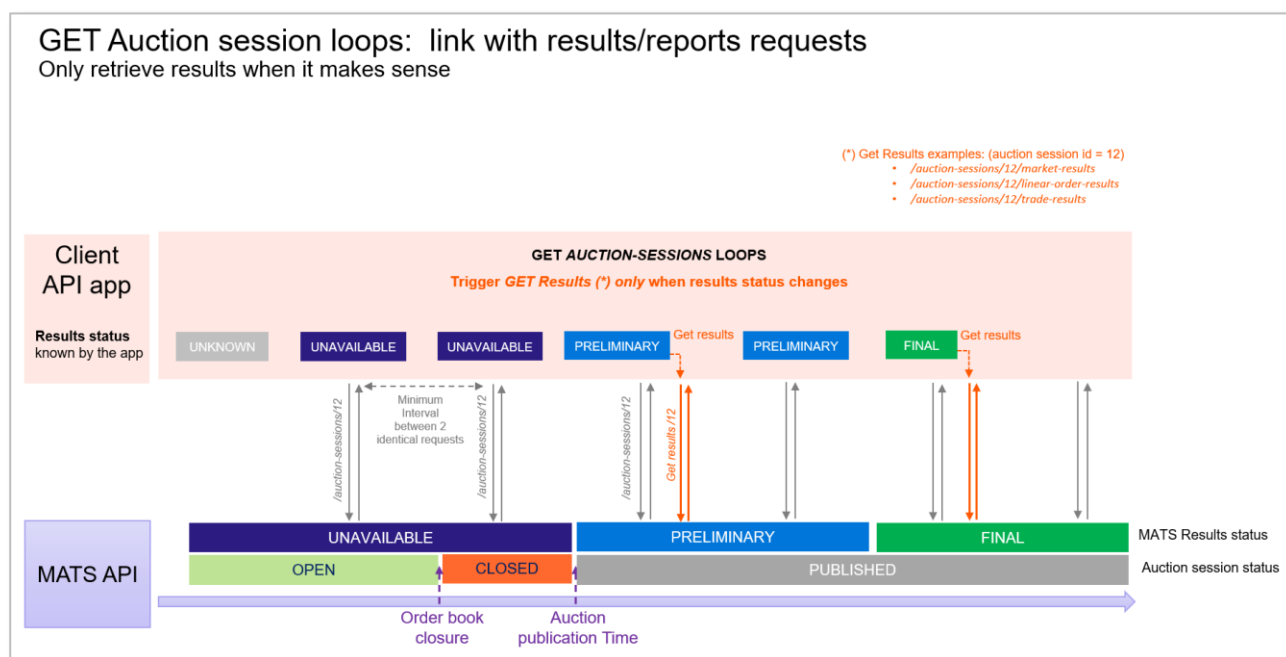
3.1.1.3 Auction-sessions / Market Results policy

With MATS, auction sessions results status(es) is/are provided in the response of *auction-sessions* requests : in order to detect the status (OPEN, CLOSED, PUBLISHED, CANCELLED) and results status (e.g. UNAVAILABLE, FINAL) changes/transitions, API apps which do not implement the Push must:

- loop on Get *auction-sessions* requests, respecting the above interval between requests,
- send a set of results/reports request(s) when the auction status and results status changes from a response to another, indicating that new results are available and should be retrieved.
 - for instance:
 - *CLOSED / UNAVAILABLE* → *PUBLISHED / PRELIMINARY*
 - *CLOSED / UNAVAILABLE* → *PUBLISHED / FINAL*
 - *PUBLISHED / PRELIMINARY* → *PUBLISHED / FINAL*

Market result statuses sequences depends on the nature of the auction:

- Day-ahead and intraday local auctions:
 - *Unavailable* (Not yet published) -> *Final*.
- Pan-IDAs coupled auctions
 - *Unavailable* -> *Preliminary* -> *Cancelled*
- Day-ahead and intraday coupled auctions:
 - *Unavailable* -> *Preliminary* -> *Final*
 - *Unavailable* -> *Preliminary* -> *Cancelled* -> *Unavailable* -> *Preliminary* -> *Final*



3.1.1.4 If you are using the Push

If Push is implemented, then you are allowed to optimize your apps as follows:

- Listen to push notifications and based on the notifications you received send the respective GET requests
- As the push message delivery guarantee is **at most once** (please refer to the Push *MATS Push Client Application Design Guide* for more details), you are allowed to implement loops for GET request(s) to complement the push feature. These loops must respect our **ToR**.

3.1.2 Requests validation

In general, the MATS API will validate messages. Invalid messages will be rejected.

If an application is erroneous and sends invalid messages in short frequency, this could have the effect of a denial of service attack and can result in a suspension of the related application.