AI-ENABLED SOFTWARE TESTING FOR **ARTEX MTF**

ARTEX–Exactpro Case Study

| ARTEX GLOBAL MARKETS BACKGROUND

Throughout history, interest in art investment has remained unwavering. However, the ability to take ownership and enjoy the aesthetic and commercial benefits of art has, so far, been a privilege of the select few. ARTEX Global Markets, a multilateral trading facility (MTF) for art shares, aims to democratise the art market by introducing the practice of shared ownership by many. ARTEX has set out to provide a new underlying infrastructure to enable wider accessibility to some of the world's greatest masterpieces to potential investors and bolster the attractiveness of art as a liquid asset.

ARTEX was co-founded in 2020 by art enthusiasts and financial market experts H.S.H. Prince Wenceslas of Liechtenstein and Yassir Benjelloun-Touimi. It is regulated and supervised by the Financial Markets Authority of Liechtenstein within the European MiFID II legislative framework. The art listings are expected to span a period from the Renaissance to the twentieth century. It is anticipated that ARTEX-listed masterpieces will be on public display in museums and exhibitions around the world. The MTF aims to list more than EUR 1bn worth of artworks in the coming quarters.

The ARTEX MTF offers a continuous trading model with auctions. Trading starts with an opening auction, it is followed by continuous trading via a central limit order book and ends with a closing auction.

| INTRODUCTION

To ensure fair and orderly trading and establish effective contingency arrangements to cope with risks of systems disruption, ARTEX partnered with recognised exchange infrastructure providers and engaged Exactpro for comprehensive testing of its trading venue.

The case study highlights Exactpro's **AI Testing** approach that was tailored to ARTEX needs and encompassed end-to-end functional and non-functional testing of the MTF's protocol and matching engine software. The case study also outlines the importance of early testing and effective inter-team collaboration.

Engaging the independent AI-enabled software testing expertise of the Exactpro team has helped ARTEX elevate the quality and operational resiliency of the MTF infrastructure and enrich the firm's growing testing practice with a diverse set of tools and perspectives into various aspects of its underlying technology.

Alexandre Reynaert,
Chief Technology Officer, ARTEX Global Markets,
comments on the ARTEX–Exactpro collaboration:

"Building a new trading venue is not a lightweight exercise. It was of paramount importance to guarantee a flawless experience for our members and participants. Exactpro's insights & experience were invaluable and material in ensuring that our trading venue was robust, consistent and compliant with the industry standards.

"During our common testing phase, the ARTEX MTF was not launched yet. It allowed us to have a fast-paced continuous deployment and integration, only following a light change management process. The Exactpro teams easily sustained our continuous deliveries and feedback. Our collaboration was very agile with cycles of less than 24 hours on average. The dynamic created between our respective teams was extremely positive and beneficial.

"Exactpro's ability to adapt and update their test libraries & models to our changes was outstanding. Their use of AI to reduce the size and execution time of the testing process, without compromising the quality of the testing, was inspiring. There was also great value in the End-to-End testing aspect to ensure consistency and detect regressions.

"Exactpro were highly knowledgeable with a great attention to detail. Their testing rounds and reports gave our partners, participants, and stakeholders a peace of mind prior to our launch. On March 8th, 2024, ARTEX successfully listed the first artwork (Francis Bacon, "Three Studies for Portrait of George Dyer", 1963, LU2583605592) and has been running incident-free since then."

Iosif Itkin,
co-CEO and co-founder, Exactpro,
comments:

"ARTEX has been an exemplary implementation of the shift-left testing concept. The software testing team was brought on board early on and got to guide the development team to the necessary level of complexity required from this type of financial infrastructure.

"Having invested in independent software testing at this early stage, ARTEX was able to avoid a lot of the common pitfalls, resolve defects early and at a lower cost, as well as significantly increase their overall delivery speed. Now the ARTEX trading venue is supported by an extensive test library that can be used regularly for regression testing. Being easy to run and update, the regression library also significantly facilitates the introduction of any future code changes, should ARTEX choose to expand.

"We very much appreciate the trust put in Exactpro in this collaboration and have enjoyed the opportunity to share our industry expertise and showcase our AI-enabled software testing approach."

exactpro

❚ AGILE DELIVERY AND COLLABORATION

The Project spanned 4 phases.

Phase 1 focused on the initial exploration of the system under test (SUT), building its digital model ('digital twin'), and detecting and reporting a major portion of defects and change requests. The testing and development teams stayed in close contact, to reduce feedback loops. The testing team received a new version of the system, on average, every other day. The system had significantly improved by the end of Phase 1, but a number of known open defects remained, which the ARTEX team tried to rectify before the start of the next phase.

Phase 2 followed ARTEX's review of the test evidence previously put forward by Exactpro and focused on retesting the defects fixed by ARTEX and covering the new functionality. Most of the changes in scope of the second test phase were related to the Volatility Control Mechanism (VCM), Quotes and Reject Codes. By this phase, the SUT's non-functional aspects such as stability and performance had significantly improved. Improvements in the system's determinism allowed Exactpro to speed up test execution and increase the diversity of test scenarios by 50%.

Phase 3 took place after code changes focused around performance and capacity improvements that were not directly related to the main functionality. These, however, caused regression in Quotes and Orders matching. It also resulted in a resurgence of the latency problems resolved between phases 1 and 2. Having a well-shaped test library previously produced by Exactpro enabled the regression defects to be detected and fixed within a short timeframe.

The test library had to also respond to the latest changes of disabling the Self-Match Prevention (SMP) functionality. The mechanism – initially embedded into the ARTEX platform – prevented self-trading of orders entered with the same user ID or member. This was later deemed an impediment, considering that the market participants trading on the MTF are individual investors represented by a single broker. Monitoring mechanisms were put in place, instead of the SMP. Additional performance and capacity testing were executed prior to the system's go-live.

The dynamics of the created vs resolved defects are illustrated in **Fig. 1**.

Phase 4 focused on integrating the automated regression testing solution into the ARTEX CI/CD pipeline for continuous internal E2E regression checks against multiple environments, with stand-by analytical and technical support by Exactpro in the post-integration period.

Investing in the effort of creating a unified regression test library, ARTEX has pursued several goals:
1) Obtaining a **fully automated**, **efficient**, and **easy-to-run continuous regression testing tool** for daily checks and checks in-between software versions;
2) Gaining the ability to perform **functional testing under load** for any of the following versions of the SUT, provided no major changes to code are introduced;
3) Obtaining the ability to run the test library against **multiple environments**.

The ARTEX regression testing library is a full-scale environment connected to market data, market surveillance and other real-time feeds, with the testing tool fully simulating the trading day schedule and verifying the correctness of the system's work during the pre-trading, opening and closing auctions, and regular trading phases. With such a set-up, multitudes of tests could be run in parallel and against numbers of instruments simultaneously.

As part of the ARTEX–Exactpro engagement, ARTEX is to receive version upgrades and other improvements in the testing methods and the integrated toolset.
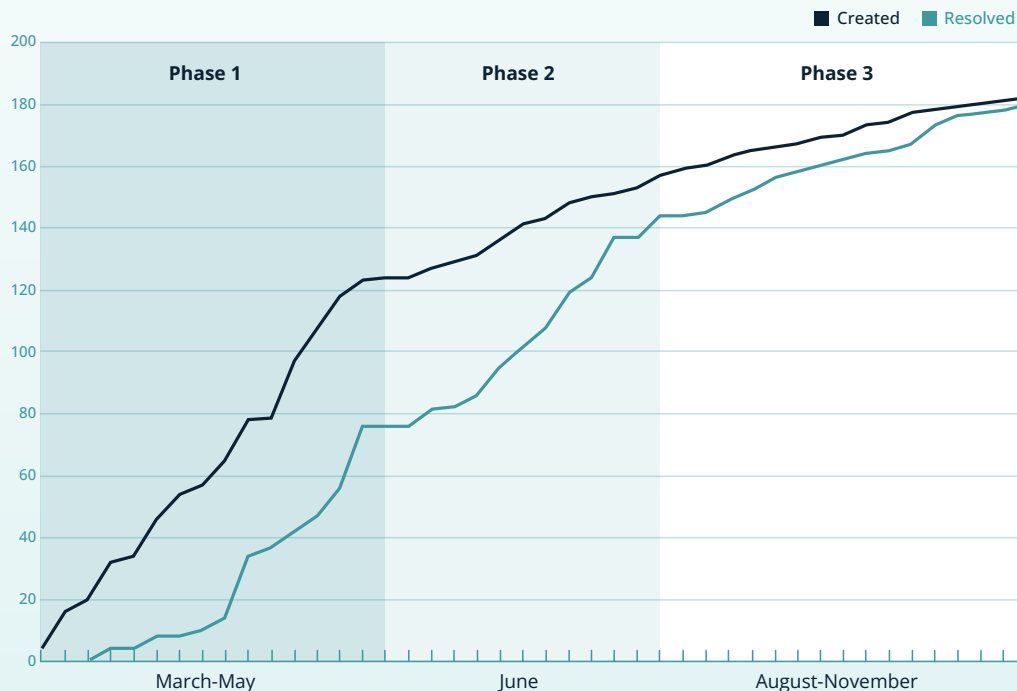

Fig. 1 *Created vs resolved defects throughout Phases 1–3*

PROJECT SCOPE

The testing scope encompassed the following activities:

- **Functional testing**: FIX connectivity, FIX message tags, ARTEX matching engine functionalities in the following areas: Auction, Amend, Cancel, Daily life cycle (DLC), Mass Cancel, Matching, Market by Order (MBO), Market by Price (MBP), Quotes, Rejects, VCM.
- **Non-functional testing**: latency, throughput and capacity measurements performed under steady load generated during functional test execution, to ensure exhaustive coverage of the SUT's performance aspects. This Exactpro approach is called 'Functional Testing Under Load.'

Out-of-Scope areas included:

- Predefined reference data setup/market operations
- Some of the reject codes (those not related to business scenarios)
- Failover scenarios
- Integration testing

exactpro

TESTING SCOPE LIMITATIONS

As is the case with many financial system projects, not all SUT functionality is available to a third-party software services provider. Over the course of the Project, the Exactpro team has had no access to the following:

- **Backend** – the active testing approach was the main source of information for model development (to learn more about active and passive testing techniques, see Exactpro's Trading Technology Testing case study).
- **Logs and monitoring** – the data from these sources was not included into the list of coverage points.

Other Project constraints included:

- **Capacity limitations** – load testing and throughput measurements were performed in a limited mode due to system access restrictions.
- **Uncross trading capacity limitations** – a limited number of scenarios related to uncross during auctions could be run per day.
- **Time constraints** – to match ARTEX's agile development pace, the team needed to reduce the timeframe required for a full regression testing run while keeping the quality of the test coverage.

Trading Technology Testing with th2

AI TESTING

To address the above scope limitations and project constraints, the Exactpro team implemented a test generation and test library optimisation approach combining model-based testing (MBT) and the power of artificial intelligence (AI) algorithms. Using a digital model of the SUT and enhancing it with sets of AI algorithms allowed the Exactpro team to:

- explore system behaviour without the need to access the system's source code;
- have the test framework fully match the complexity of the SUT;
- achieve exhaustive – yet resource-efficient – test coverage by producing a minimal acceptable amount of realistic feature-rich scenarios simulating authentic trading participants' actions;
- continuously improve the quality of the test library over the three testing phases by fine-tuning the model's accuracy and generative capabilities.

The AI-enabled model-based approach offered by Exactpro has complemented the in-house quality assessment mechanisms established at ARTEX and enriched its system exploration capacity.

The AI Testing approach implemented at ARTEX consisted of the following steps illustrated in **Fig. 2**. We also break down the steps of the approach in this section.
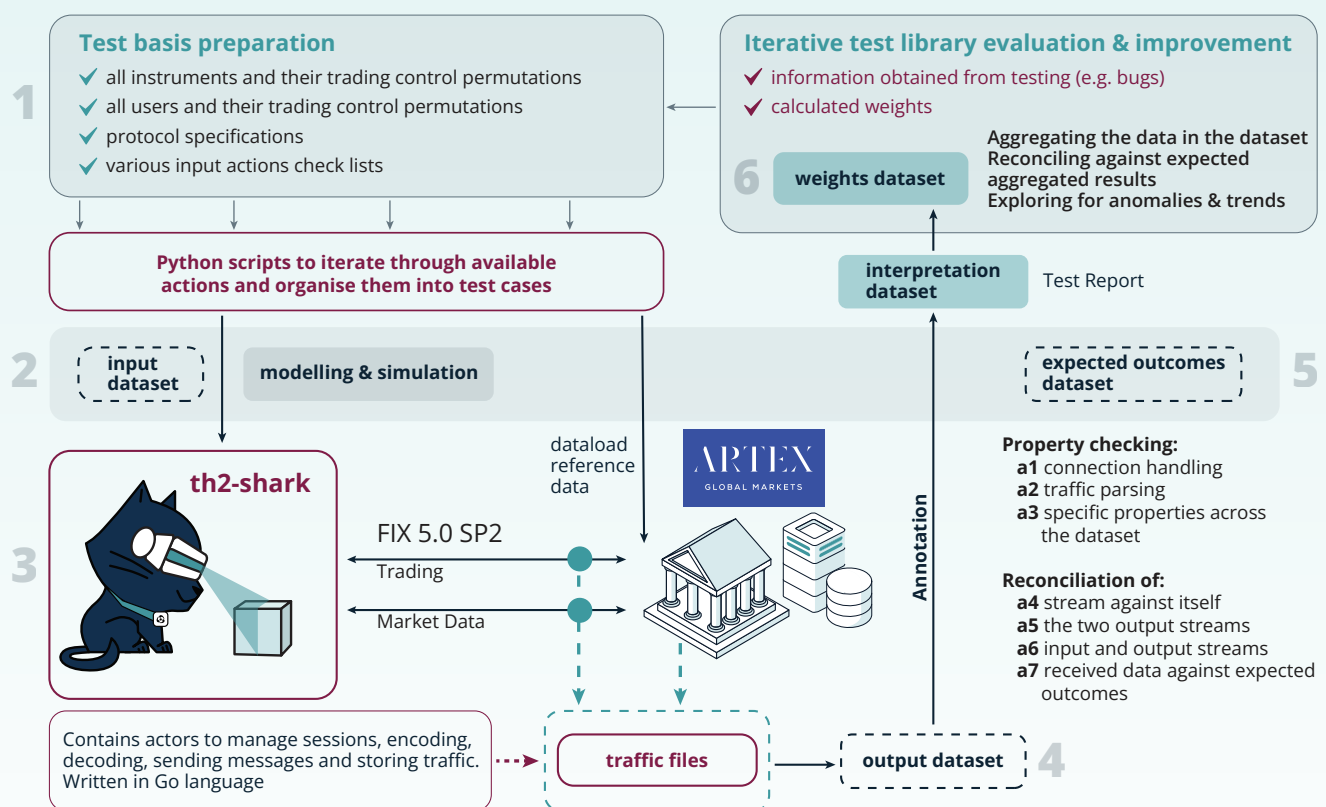

Fig. 2 *The Exactpro approach to Test Library Generation and Optimisation with AI*

exactpro

STEP 1. TEST BASIS ANALYSIS

The modelling task on the project relied on Exactpro's proprietary global exchange model being customised in line with the test basis provided in the ARTEX team's requirements, interface specifications, and other documentation artefacts describing the SUT (collectively known as the test basis). Step 1 of the approach featured the analysis of the information available and its conversion into machine-readable form.

STEP 2. INPUT DATASET & EXPECTED OUTCOMES DATASET

The knowledge derived from the test basis laid a foundation for creating a **'digital twin'** of the ARTEX trading venue – a machine-readable description of its behaviour expressed as input sequences, actions, conditions, outputs, and flow of data from input to output. This model serves as the basis for **test coverage point** discovery. Coverage points are the elements of the test basis that characterise system behaviour. Information extracted from the test basis is used to produce **source test** scripts – test execution instructions formalised in machine-readable form.

To account for a multitude of variable data permutations intrinsically suggested by the functional capabilities of the ARTEX trading venue, Exactpro's proprietary model generated an extensive test library by modifying the contents of the source test scripts and iterating through system-specific data points. For AI training purposes, all possible modifications were transformed into an **input dataset**. The dataset used for shaping the test input included:

- all instruments and their trading control permutations
- all users and their trading control permutations
- all relevant reference data actions
- protocol specifications
- various input action checklists

This step was also used to start shaping the **expected outcomes**. This dataset is a result of running the AI-generated test scripts against the SUT model, and it essentially consists of modelled predictions of expected system outputs. The 'expected outcomes' dataset is used in Steps 4 and 5 for comparison of actual system behaviour against the expected results and for further model refinement.

STEP 3. TEST EXECUTION. OUTPUT DATASET

After the Exactpro team aggregated all possible test script modifications, they were passed on to th2-shark – Exactpro's AI-enabled framework for automation in testing – for execution.

th2-shark converted the test scripts into FIX messages and injected them into the SUT. This stage of the process was used to send as many messages as possible, considering the technical limitations: the data obtained from message injection is an important source of knowledge about the behaviour of both the SUT and the Exactpro model. Upon test execution, th2-shark collected traffic files for analysis in a raw format to compile the **output dataset**. Unified storage of such test execution data in a single database enables better access to test evidence and maximum flexibility for applying smart analytics.

STEP 4. ENRICHING OUTPUT DATASET WITH ANNOTATIONS

To subsequently apply AI-enabled analytics, the Exactpro team started attributing annotations (the annotation steps are marked on the schema as 'a1' - 'a7') to the 'output' dataset. This also laid the groundwork for the **interpretation dataset** – the dataset containing aggregate data and annotations from the processed 'output' and 'expected outcomes' datasets in Steps 4 and 5.

Annotation is a multi-step process that becomes more extensive as we obtain and interpret more information about the SUT. First, output data is marked up to reflect the **connectivity status** (marked as 'a1' in Fig. 2) between the testing tool and the trading venue. If there is no response from the SUT, the team is notified about the connection failure. Otherwise, we can proceed and add labels indicating our ability to parse the output traffic. **Traffic parsing** (marked as 'a2' in Fig. 2) ensures that we can read the received output data; it is labelled for the types of messages covered, flagging extra and missing messages, validating duplicate or absent orders on the book, marking important/ancillary messages.

Annotation is also an important feature supporting traceability – the ability to establish links between the requirements and the test results, prove that required tests have been completed and failures investigated. In a model-based approach and with non-functional testing complementing the functional checks, traceability is non-linear: a single requirement is likely to be covered by a multitude of tests performed under various conditions. The context of a complex data-driven environment makes annotation ever more important. In the regulated space, it significantly facilitates such crucial aspects of software delivery as data governance, compliance and regulatory reporting and helps foster a culture of data accountability.

Further into **Step 4**, the results received from test execution in the form of the **output dataset** continued to be annotated as the data was being reconciled against various properties (marked as 'a3' - 'a7' in Fig.2).

Examples for these steps would be assigning specific annotations whenever any of the following checks are not satisfied:

exactpro

- Each inbound FIX message receives a response
- Tag values are consistent in inbound and outbound messages
- There are no contradictions between the data in Market Data by Order and Market Data by Price channels
- Published Order Books are only being crossed during auctions
- Execution Reports in FIX order management traffic across all participants match FIX Market Data

Python source code used to perform the above checks was also used to provide branch and statements code coverage data as additional annotations/coverage points for the executed scripts.

The contents of the 'output' dataset were also compared against the **expected outcomes** dataset created in Step 2. This truly revealed the value of two separate teams (the in-house development and testing team, as well as the independent testing team) providing deliberate assessments of the quality of the SUT. Any discrepancies between the expected and the observed behaviour found at this stage are to be considered issues/defects and need to be investigated, either on the side of the model or on the side of the SUT.

STEP 5. MODEL REFINEMENT

This is the step where a major part of the development of the 'digital twin', property and reconciliation checks took place: in iterative movements between Steps 4 and 5, the model logic was gradually trained and improved to provide a more accurate interpretation of the 'output' dataset and to produce updated versions of 'expected outcomes' and 'interpretation' datasets.

In this iterative annotation process, the team also produced aggregated annotations to simplify the work of the business analysts and software testers.

As a result of iteratively running the test script permutations against the ARTEX platform and refining the model, we received a **test coverage** report for the entire test script library and were also able to highlight all coverage points in every individual test script.

Exactpro's coverage points technique makes it evident that, while each generated test script or test script part bears its unique footprint, it may be far from unique in terms of its test coverage value, which means there is room for optimisation in the volume and structure of the test scripts. Due to time and technology constraints often accompanying software delivery – particularly, in financial technology – efficient optimisation is key in creating a comprehensive end-to-end test library.

STEP 6. TEST REINFORCEMENT AND LEARNING. WEIGHTS DATASET

In pursuit of the most optimal version of the final regression library, we needed to detect the best-performing test scripts and reduce the volumes of tests required for execution to the minimal amount that, at the same time, covers all target conditions and data points extracted from the model. To do this, the Exactpro team applied discriminative techniques to 'interpretation' datasets.

Step 6 is an iterative process that requires several consecutive end-to-end test runs and subsequent analyses and refinement of the 'input' dataset and the annotation process. Each time, based on the weights attributed to them via annotations, test scripts were compared to each other in terms of their test coverage value. This helped shape the **weights dataset**. Using insights from the 'weights' dataset enabled the Exactpro team to receive AI-driven recommendations on:

- the areas requiring human attention,
- the best-performing scenarios,
- automatically pinpointing vulnerable test coverage areas that require more checks.

This saved time spent on triaging test results and deciding which 'less effective' test checks to discard. It also allowed the team to further explore the system data for anomalies and trends.

To achieve the final version of the regression library, select scenarios were fine-tuned until the Exactpro team reached the maximum possible level of test coverage and the minimal reasonable number of checks to execute. Carefully selecting and optimising a subset of test scripts results in achieving a significantly more performant and less resource-heavy version of the initially generated test library.

The final test coverage report helps demonstrate **traceability** between the requirements and the test outcomes all the way to the assigned issue ID, where applicable (in case of a 'failed' test case, it is possible to see the issue created and view its resolution history). All test report results can be traced back to the requirements and vice versa – the requirements are linked to all test cases relevant for their coverage.

exactpro

❚ AI TESTING TAKEAWAYS

As mentioned above, **the regression library developed by Exactpro comprises a carefully selected and optimised subset of test scripts that is a more performant and less resource-heavy version of the initial test library**. Because the test set covering the SUT had not expanded since Phase 2 (due to the need to disable the SMP functionality in Phase 3), the test library optimisation logic is best showcased with the Phase 1 vs Phase 2 numbers. Please refer to **Fig. 3** demonstrating the results of the ARTEX test library improvement, from the initially generated 'input' dataset to the regression test library delivered in Phase 2 of the Project.
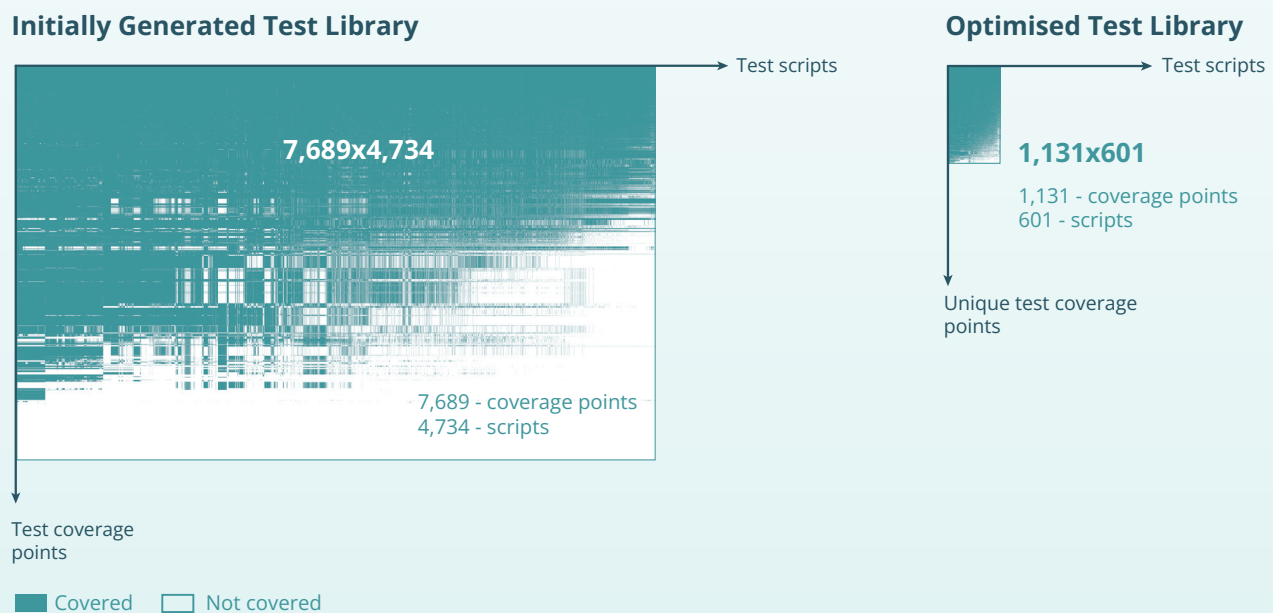
Initially Generated Test Library

Optimised Test Library

7,689x4,734

7,689 - coverage points
4,734 - scripts

1,131x601

1,131 - coverage points
601 - scripts

Unique test coverage points

Test coverage points

▉ Covered   ☐ Not covered

Fig. 3 The volume of the test library before and after the AI-enabled optimisation

This is the result of incremental improvement making the test scenarios more unique (covering rare combinations of conditions), lean and higher-performant (featuring many unique coverage points simultaneously), while keeping their volumes to a reasonably low minimum. Please refer to Exactpro's **AI-enabled Test Library Generation and Optimisation demo** for a more detailed description of the optimisation methodology.

While the size of the test library got significantly reduced, the extent of test coverage has increased over time, with the number of resources required to run the library becoming considerably lower, following the optimisation. **Fig. 4** showcases a comparison between the regression test library created at the end of Phase 1 and the improved test library delivered in Phase 2 of the Project.

**AI-enabled Test Library Generation
and Optimisation demo**

Phase 1

Test scripts →

744x312

744 - coverage points
312 - scripts

Unique test coverage points

Phase 2

Test scripts →

1,131x601

1,131 - coverage points
601 - scripts

Unique test coverage points

■ Covered   □ Not covered

Fig. 4 *Continuous AI-enabled regression testing library optimisation
(libraries delivered in Phases 1 and 2, respectively)*

Even though the Phase 2 library is clearly larger and requires slightly more resources to run, it is still a ≈**87**% reduction in the volume of test cases and a ≈**85**% optimisation in the number of test coverage points, compared to the initially generated version of the library [refer to the 'Before' section of the **Fig. 3** graph]. The Phase 2 library is significantly more versatile in terms of test coverage, thanks to the continuous AI-enabled improvement feature.

Alexey Pereverzev, Director, Exactpro Systems Limited, comments:

"Creating a software testing library from scratch is not a trivial task. Thankfully, working with a system's 'digital twin' enables us to transfer years of experience from project to project, imprinting our collective knowledge on the model each time. With each new project, we get to focus our efforts on fine-tuning the model to the specifics of a given trading venue implementation.

"The ARTEX team has been highly responsive to the issues we raised and highly agile in delivering new releases. In a very limited timeframe, we have created a full automated regression library and iteratively enriched it with more versalite test scenarios and more verifications as the system matured. The extent of test coverage and the level of optimisation of the regression test library we have managed to achieve on the Project is also quite remarkable. It would have been impossible to get to this result without resorting to the use of AI-based automation on several of the testing steps."

exactpro

❙ NON-FUNCTIONAL TESTING

In line with the Exactpro 'Functional Testing Under Load' approach, select messages of the functional test library can be used to load the system and explore its performance aspects. Since most systems are not designed to have unlimited capacity, finding a failure point is a question of 'when', rather than 'if'. Once failure points are identified, we know the circumstances that cause them and know exactly how the system behaves in response to the outages. The Exactpro team has identified such points for the ARTEX infrastructure, which enables the ARTEX team to efficiently manage their resources and makes them well-informed in planning for future growth. Another reason to measure system capacity is latency and throughput calculation. Once a system reaches its peak load and exceeds its throughput capacity, it should organise incoming messages in a queue – a mechanism characteristic of all high-load high-availability systems.

LATENCY AND THROUGHPUT TESTING. LATENCY ANALYSIS FOR DISTRIBUTED STORAGE

Round-trip Latency is the time it takes for a client's message to reach the trading venue and get acknowledged by it, i.e. to receive a corresponding report back from the trading venue. **Throughput** measurements aim to establish how many messages per second the system is capable of receiving and processing. From the perspective of order matching, ARTEX is a queueing network system – a type of system where separate queues are connected by a routing network on the first-come, first-served (FCFS) basis.

The principles of work of queueing network systems are formulated by the Queueing Theory – a mathematical study of waiting lines, or queues. The heart of such a routing network is the queueing node consisting of one or several servers. Requests arrive to the queue, await their turn, take time to process, and then depart from the queue. When a request is completed and departs, the server that processed it becomes available to be paired with another incoming message. A queue typically has a waiting zone of up to **n** incoming messages and a buffer of size **n**. **Fig. 5** illustrates queuing logic of a trading infrastructure:

Using Queuing Theory, the Exactpro team was able to use round-trip latency calculations to quantify the system's throughput **at and above its processing capacity**.
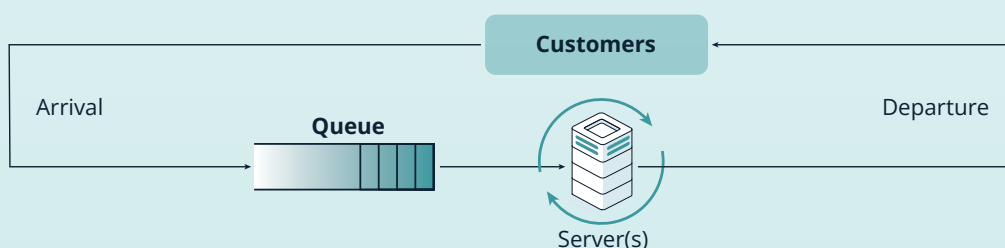

Fig. 5 *Queuing logic of an order matching system*

time series latency chart
latency, ms

latency, ms

X mps

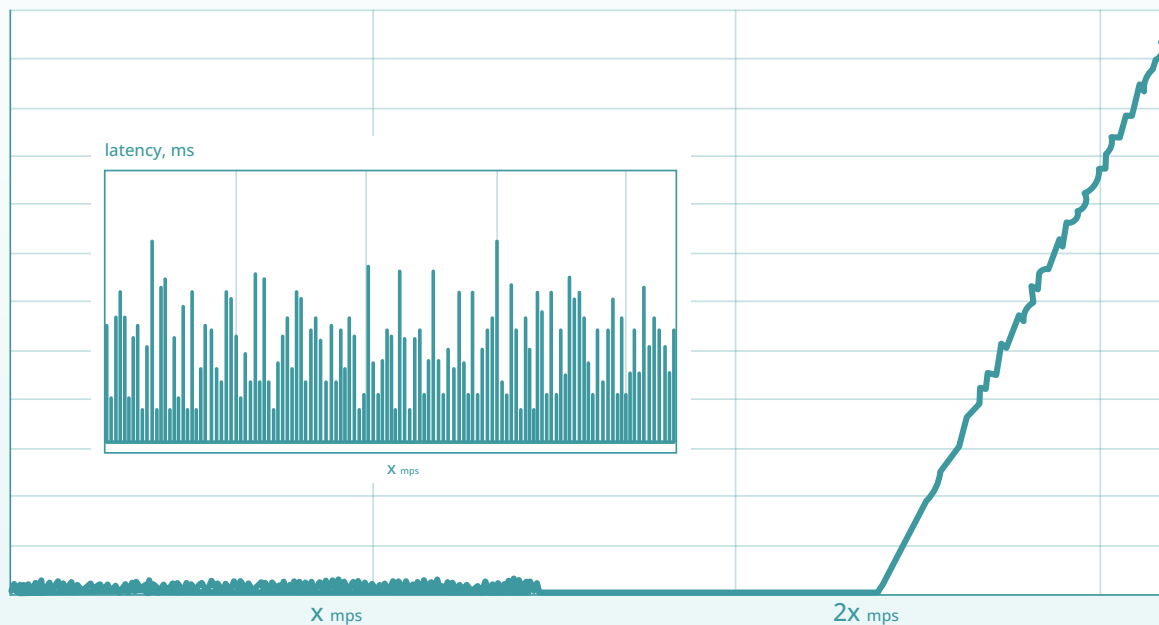X mps   2x mps

Fig. 6 *Time series latency chart*

The system's maximum throughput can be determined by examining a time series graph to identify at which load rate the round-trip time starts to increase. In the time series latency chart presented in **Fig. 6**, X displays the message rate per second and Y shows the latency for a given second. The numerous spikes you can see on the graph are the manifestations of the SUT's queuing process. The length of the latency chart helped us obtain two parameters: T1 and T2 (the time we started sending the messages, and the time we stopped, respectively) which were used to calculate the SUT's throughput.

It is important to also visualise latency distribution across the entire set of messages injected, this was done by plotting a frequency distribution graph for the ARTEX platform's latency measurements. In the resulting latency distribution graph, the X axis displays the occurrence of the metric value (milliseconds), and the Y axis displays all the values of the specified metric (number of samples/messages) in ascending order. In the **Fig. 7** chart, we see results for the extended capacity test.
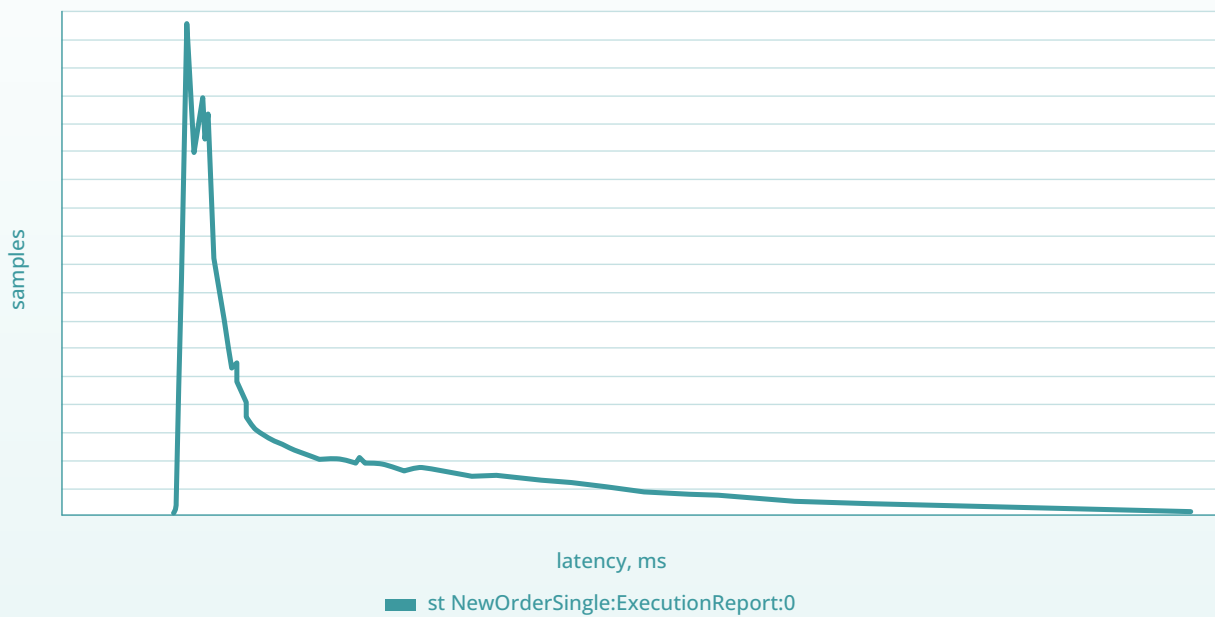
exactpro

Fig. 7 *Total ramp-up test round-trip latency distribution at throughput processing capacity*

Frequency distribution graphs are useful for obtaining an accurate estimate of the system's most probable response time to messages.

CONCLUSION

The case study has demonstrated Exactpro's AI Testing approach implemented for the ARTEX Global Markets infrastructure. The approach is based on the principles of E2E testing and model-based testing, it is enhanced with AI-enabled processes and spans functional and non-functional testing of the ARTEX protocol and matching engine software. The case study also brings out the importance of early testing and effective inter-team collaboration.

Leveraging a combination of system modelling and AI-enabled software testing in verifying the quality of financial technology systems helps transform the quality and size of the test libraries to achieve better efficiency, provide more effective test coverage of the system's functionalities – resulting in better detection of potential vulnerabilities – and introduce advanced levels of automation. With the main analytical power siloed away from the code, the Exactpro AI Testing approach applies to a wide variety of business use cases within the transaction-based systems space, regardless of their scale and maturity level. Models are fine-tuned to test and build systems with similar functionality and enable software delivery at a much faster pace, compared to conventional approaches.

Engaging independent AI-enabled software testing as early as possible in the product's lifecycle allows stakeholders to receive objective information about the system at the outset and resolve most issues before they become critical.

Due to its ability to unlock greater control over system components and functions, AI-enhanced system modelling provides a fundamentally deeper extent of system analysis for numerous business cases, strengthening the operational resiliency of the finance ecosystem.

exactpro

EXACTPRO AT A GLANCE

Exactpro is an independent provider of AI-enabled software testing services for financial sector organisations. Our clients are exchanges, post-trade platform operators, and banks across 20 countries. Our area of expertise comprises protocol-based testing of matching engines, market data, market surveillance, clearing and settlement systems, payments APIs. We help our clients to decrease time to market, maintain regulatory compliance, improve scalability, latency and operational resiliency. Exactpro is involved in a variety of transformation programmes related to large-scale cloud and DLT implementations at systemically important organisations.

Founded in 2009, the Exactpro Group is headquartered in the UK and operates delivery centres in Georgia, Sri Lanka, Armenia, Lithuania and the UK and representative offices in the US, Canada, Italy and Australia.

Exactpro was part of the London Stock Exchange Group (LSEG) from May 2015 until January 2018, when the Exactpro management went through the buyout of the company from LSEG.

CLIENT NETWORK

Amsterdam
Oslo
Stockholm
Montréal
London
Zurich
Toronto
Greenwich
Vienna
Paris
Tokyo
New York
Milan
Vaduz
Astana
Jersey City
Madrid
Athens
Osaka
Abu Dhabi
Hong Kong
Doha
Bangkok
Lagos
Colombo
Johannesburg
Singapore
São Paulo
Sydney
Cape Town