

AI TESTING: DELIVER BETTER SOFTWARE, FASTER

Software testing is an information service. Its goal is to provide stakeholders with objective information about the defects present in their system. A software defect is anything in the code, configuration, data or specification that can decrease the value of software to its stakeholders. Building and operating software without thorough testing is like devising a military strategy without proper intelligence, treating patients without accurate diagnostics or participating in electronic trading without access to market data.

The effectiveness of an information service can be judged based on its accuracy, relevance and accessibility. Improving software testing implies making it progressively better at detecting and interpreting defects, whilst reducing the timeframes and costs.

With the growth of available digital data and computational capabilities, we are seeing the use of subsymbolic AI deliver improvements in autonomy and efficiency across many industries. In software testing, using AI can help harness the power of big data analytics to enhance the generation of test ideas and the interpretation of test results. Smart execution capabilities can support full-scale AI adoption in software testing.

Do you think you can benefit from obtaining important information about your software earlier, at an affordable cost?

We are all good – we have teams and procedures in place to take care of all our testing needs.

Applying **data analytics** to various levels of an existing process enables us to boost its effectiveness. A continuous improvement cycle is created as a result of **AI integration**.

The better a process is organised and is performing, the more it can be improved with AI testing. Process deficiencies can be detected and rectified in an ongoing improvement loop.

Our system is too complex and specific, approaches developed for other systems or domains will not work here.

Complex transaction processing systems share similarities: they have to take into account a wide variety of configuration data and to process lots of transactional data, high resilience (availability) requirements and a complex operational life cycle are also commonplace among them. The **data annotation** process may vary for different types of such systems, but the rest of the data processing can be the same.

Breakthroughs tend to happen at the confluence of different disciplines. Any individual team can benefit from the latest technology achievements and use expertise from related domains to its advantage.

Your proposal is too complex for our people.

Experiments and training in **machine learning**, **data processing** and **software testing** can amplify your team's expertise and add value to aspects of your organisation's work that go beyond software testing.

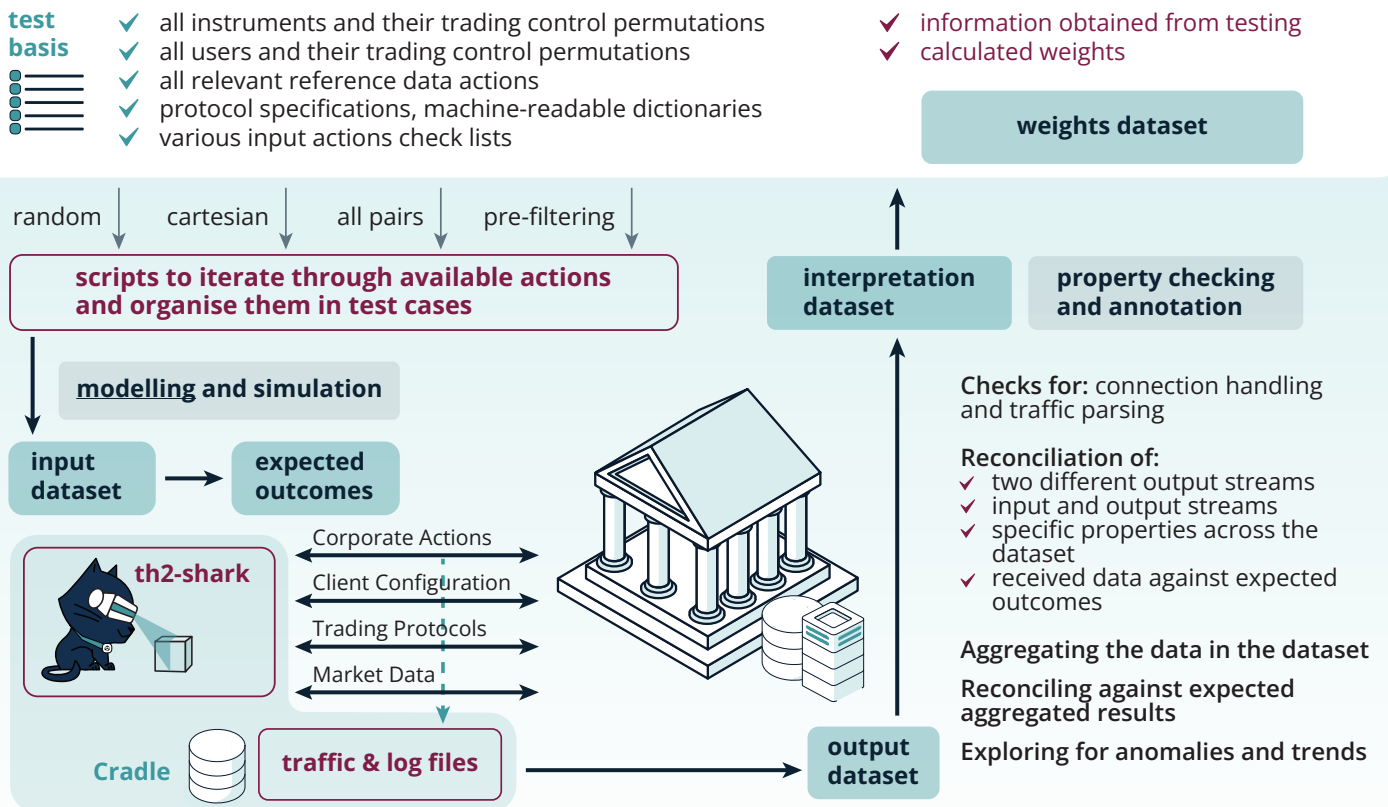
You never know what you are capable of until you try.

We are too early in our journey, our development has just started.

Early-stage system modelling and **data analysis** are processes that positively affect architecture design and development of your application. Alignment between development and testing is key in delivering better solutions faster.

Starting testing as early as possible helps secure a solid foundation, it also ensures a smooth development process and product quality by design.

Case Study: Machine Learning In Exchange Testing



Automating Cognitive Functions in Software Testing

Automation is the technology by which a process or procedure is performed with minimal human assistance. Software testing is a process of empirical technical exploration of the system under test that consists of selecting a subset of possible actions, executing them, observing and interpreting the outcome. With modern CI/CD pipelines, it is easy to minimise human involvement in the execution part. Automating higher-level cognitive functions such as test ideas generation or results evaluation is, of course, a much less straightforward task.

Test basis is the body of knowledge about the system used for test analysis and design. The AI testing process includes the following steps:

- Converting parts of the test basis into machine-readable form and annotating the obtained data for **Supervised Learning**.
- Using synthetic test data generators to produce substantial datasets.
- Improving and optimising the input dataset with **Reinforcement Learning**, based on the extracted features.
- Running arbitrarily many computer-assisted checks and applying scripted rules and **Unsupervised Learning** to extract features from the output dataset.

Shorten the feedback loop and use the obtained information to build competitive products and solutions.

If you want to know more about leading industry expertise in software testing or are looking to leverage AI to boost your software development processes, reach out to us via exactpro.com.

