

Ensuring Confidence in Trial Simulations: An Automated Testing Pipeline

Anoop Rawat and Hema Patil, Cytel Statistical Software & Services Pvt. Ltd.

Introduction

East Horizon performs core statistical computations required to evaluate complex clinical trial designs, including:

- Test statistics and decision rules
- Power, Type I error, and other operating characteristics
- Interim and final analysis outcomes for adaptive and complex designs

To support deep design understanding, the platform provides rich analytical capabilities for result exploration:

- Interactive tables and visualizations
- Filters and scenario comparisons

As new trial designs and enhancements are continuously added, it is critical to ensure:

- New functionality works correctly
- Existing designs and features remain unaffected

This creates a strong need for robust and scalable testing.

Why Is Automated Testing Important?

Clinical trial design platforms are inherently complex and require validation across multiple dimensions:

- Numerical accuracy of statistical outputs
- Correctness of visualizations and UI interactions
- End-to-end consistency of results

Manual testing:

- Is time-consuming and does not scale well
- Becomes impractical as the number of designs and features grows

Testing does not end at release:

- Periodic regression testing is essential to ensure future development does not break existing functionality

Automated testing enables continuous validation at scale.

Challenges in Automation

Automating testing for a statistical platform presents several challenges

• **Multiple technology stacks**

- Core computation engine
- Backend services and APIs
- Frontend UI and visualizations

• **Diverse testing requirements**

- Numerical validation of statistical results
- Visualization and UI behavior testing
- Integration and workflow validation

Each layer requires different tools, techniques, and validation strategies, increasing overall testing complexity.

How We Automated Testing?

Testing is integrated throughout the user journey while building and maintaining East Horizon features.

Testing Approach

- Automated testing is developed with feature implementation and reused for regression testing
- JavaScript and Python are combined into a unified testing framework that covers most of the application
- Context-specific testing tools are used based on the component being validated

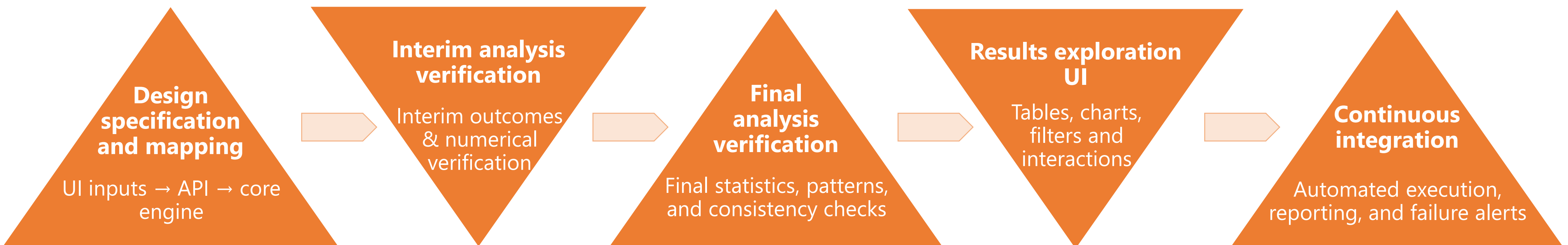
Feature Level Testing

- Multiple input scenarios are created for each new design or enhancement
- Each scenario defines:
 - Inputs (design parameters, filters, interactions)
 - Expected outputs (numerical results, tables, visual patterns)
- Tests are executed after feature development to validate outputs vs expectations

Verification Strategy

- New design algorithms are verified by subject-matter experts
- API-driven testing validates data flow and results
- UI and visualization testing is automated using Python-based Selenium
- Unit tests validate core computation logic at the component level

User Journey Coverage



Impact of Automated Testing

- Early detection of defects and inconsistencies
- Reduced risk of breaking existing features during new development
- Faster feedback cycles for developers and statisticians
- Greater test coverage with fewer manual resources
- Improved confidence in platform stability and correctness

Future Directions

- AI-enabled risk-based testing --> Prioritizing test execution based on impact and change risk
- Automated and semi-automated test planning --> Intelligent generation and maintenance of test scenarios --> Reduced manual effort in keeping tests aligned with evolving designs