

MLabs – Apropos Close-Out Report

A. Name of project and Project URL

MLabs – Apropos for Property Tests - 900151 -
<https://cardano.ideascale.com/c/idea/63770>

B. Name of project manager

Ben Hart / Mark Florisson

C. Date project started

November 2022

D. Date project completed

July 2023

E. List of challenge KPIs and how the project addressed them

Overview

Property tests help ensure mission-critical dApps don't contain vulnerabilities. They accomplish this by verifying formal properties are upheld throughout code blocks. This is opposed to unit tests which simply assure atomic pieces of code return the correct, anticipated value. For Haskell, QuickCheck is a groundbreaking and effective tool for developing property tests.

However, writing tests to check every contract permutation and edge-case state is exhausting, if not wholly impossible. Apropos is a metaprogramming toolkit for generating QuickCheck-style property tests from a logical model. It aims to make thorough testing of code simpler – a lot of the heavy lifting is automated – while ensuring edge cases are not overlooked.

Apropos relies on a logical model class to describe the properties of an application. Tied to the plutus-simple-model on the backend, Apropos leverages an SAT formula and parameterized generators to create easily tested models and their relevant property tests automatically simplifying the process of edge case testing.

F9: Developer Ecosystem

Apropos was part of the Developer Ecosystem challenge of Fund9. The challenge sought to prioritize the development of tools needed to streamline development on Cardano and attract developers to the space.

Apropos addressed the priorities of the Developer Ecosystem challenge by attempting to deliver tooling that would reduce the work of developers testing the correctness of their applications. The testing suite attempted to leverage the deterministic properties and logical underpinnings of the Haskell language to reach beyond simple property-based testing. It did so by relying on SAT solvers to thoroughly model the properties of a Cardano-based application.

F. List of project KPIs and how the project addressed them

Our Fund9 proposal looked to build out the feature set of Apropos in several ways:

- Modelling transactions and transaction graphs
- Extending script testing support
- Modelling CPU/Memory usage of scripts in relation to protocol parameters
- Documentation, examples, and how-tos
- Improvements and simplifications to API
- Other ease-of-use improvements

However, as explained in our closeout report [<https://www.youtube.com/watch?v=cRjxYWxD15s>], these objectives proved difficult to achieve. Early, development revisited a key feature, Apropos' Adjunction class. However, several issues related to our approach here eventually made it difficult to test complicated application interactions. Significant tooling – outside the scope of our original proposal – would likely be needed to improve the situation.

Despite this major disappointment, on the positive side, we were able to significantly develop a key dependency of the Apropos test suite, the plutus-simple-model test library. This included the significant achievement of incorporating Plutarch-generated validators into the library as well as their testing.

G. Key achievements

The main achievements of this project included the development of the plutus-simple-model mock chain testing library.

H. Key learnings

This experience with Apropos has demonstrated the complexity of property testing within the unique demands of the Cardano-based applications. Our experience brought to light

the nuances of testing complicated interactions and the limitation of theoretic-based applications to this domain.

I. Next steps for the product

Apropos is currently paused and it is unlikely to be restarted in the foreseeable future. Unfortunately, Apropos' lack of multi-script test support and the limitations of its logical model approach have led to a limited range of use cases and developer adoption. If restarted in the future, multi-script support would likely be the primary development goal.

J. Final thoughts

Despite the challenges of Apropos, the project presented a unique opportunity to understand the reality of theoretical applications to testing on-chain interactions on Cardano. Developments made in the plutus-simple-model library remain a positive takeaway, although our initial goals for Apropos proved challenging to achieve. Even though Apropos currently has a narrow range of use cases, the experience of this project will guide our future work and our ambitions regarding theory-based tooling on Cardano moving forward.

K. Links

MLabs website: <https://mlabs.city>

GH:

<https://github.com/mlabs-haskell/apropos-tx>

<https://github.com/mlabs-haskell/plutus-simple-model>

<https://github.com/mlabs-haskell/hedgehog-plutus-simple>