

After many months of R&D, starting today we will publish more blogposts and articles about the Juvix project and its research. Just wanted to share with the ETH research community in case there are people interested in the topic of secure smart contract design / languages.

In addition to validation and protocol development, in the past several months Cryptium

Labs has embarked upon a new project: research & development of a novel smart contract language, Juvix. Juvix is designed to address the problems that we have experienced while trying to write & deploy decentralised applications and that we observe in the ecosystem at large: the difficulty of effective verification, the ceiling of compositional complexity, the illegibility of execution costs, and the lock-in to particular backends. In order to do so, Juvix draws upon and aims to productionise a deep reservoir of prior academic research in programming language design & type theory which we believe has a high degree of applicability to these problems.

There should be a substantial bar to meet before electing to write a new language. After investigating many simpler approaches and developing distributed ledgers & smart contracts ourselves, we've decided that this bar, for the use-case of smart contracts on public ledgers, is met — there are many unique, fundamentally difficult problems which can be convincingly solved at the language level, but only by designing & engineering a language and

compiler stack from scratch.

This post, the first part of a two-part series, explains the background of considerations and requirements that motivated us to design a new language.

You can find the full article here: <https://research.cryptium.ch/the-why-of-juvix-part-1-on-the-design-of-smart-contract-languages/>