

Programming Language Implementation

The Adamnite Programming Language (name pending) should be a high level smart contract programming language. Particular attention should be given to simplicity: programming using the Adamnite high level language should be as easy as programming on C++ or Python. Particular attention should be given to modularity and API support: developers should easily be able to import libraries for use with basic smart contracts such as creating a sub currency, and interact with the blockchain directly to send automated transactions.

Another key aspect of the programming language should be safety. Type declaration should only be used when absolutely necessary, and general manipulation of the underlying state should be restricted unless absolutely necessary. A formal set of specifications will be built at the end of the research period.

Below, some resources for PL research are included:

[Brown PL Book](#)

[Another good example](#)

[Some good criticism on Solidity](#)

[A good research paper on the topic](#)

[Nick Szabo's Contract Language](#)

[And, when you are ready, Solidity' implementation](#)

Feel free to take notes, draft diagrams, and add additional research below. Also, the github repo titled A1 on Adamnite is available for public use. Please make your pull requests to the main repo as a separate branch. A branch can be dedicated to one person, or a mini-team.