# Proof-Carrying Smart Contracts

Thomas Dickerson[1], Paul Gazzillo[2], Maurice Herlihy[1], Eric Koskinen[2], and Vikram Saraph[1]

[1] Brown University
[2] Stevens Institute of Technology

**Abstract.** This is preliminary work on reconciling the apparent contradiction between the immutability of idealized smart contracts and the real-world need to update contracts to fix bugs and oversights. Our proposed solution is to raise the contract's level of abstraction to guarantee a specification $\varphi$ instead of a particular implementation of that specification. A combination of proof-carrying code and proof-aware consensus allows contract implementations to be updated as needed, but so as to guarantee that $\varphi$ cannot be violated by any future upgrade.
We propose proof-carrying smart contracts (PCSCs), putting formal correctness proofs of smart contracts *on the chain.* Proofs of correctness for a contract can be checked efficiently by validators, who can enforce the restriction that no update can violate $\varphi$. We discuss architectural and formal challenges, and include an example of how our approach could address the well-known vulnerabilities in the ERC20 token standard.