



# Function Calls as Citations

Sina Iman, Philipp Koellinger, Christopher A. Hill

DeSci Labs

## Abstract

"Function calls as citations" refers to the ability to compose a novel research object by re-using an existing component, such as code or data, from a prior research object stored in an immutable and verifiable way. This idea originates from Balaji Srinivasan, who recently issued it as a challenge to builders in the DeSci space. In this preliminary report, we answer this challenge with a proof of concept. We created two Research Objects stored on an Open State Repository, each containing code components. A new Research Object calls functions from the code components of another Research Object. Both Research Objects are stored on an Open State Repository, and interoperate using decentralized persistent identifiers.

## Open State Repository

We created a decentralized repository composed of a smart contract registry that emits events recording root hashes of content-addressed linked data objects. This repository architecture results in an "Open State Database". This means that anyone can query the registry and return the associated manifest from the IPFS network. These manifests are IPLD-compliant JSON-LD Research Objects, but the storage of any data structure is possible. The manifests contain content-identifiers (CIDs) of the linked digital resources along with their associated metadata. This forms the Research Object.

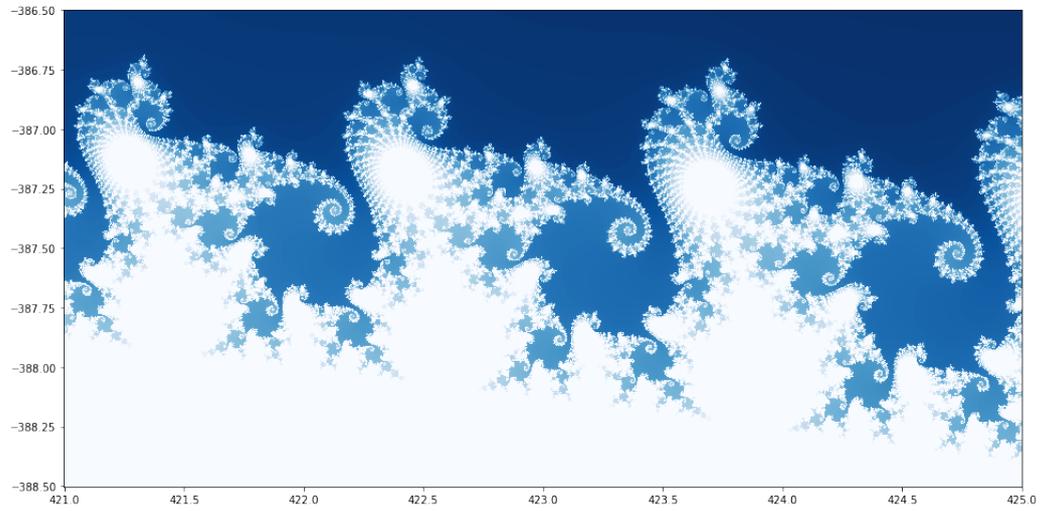
An important property of this architecture is that it mandates the generation of decentralized persistent identifiers (dPIDs). These identifiers have the property of being permanent due to the append-only nature of the implementing smart contract. There is no middleman gating their issuance: the smart contract can be invoked by any internet-connected user. When dPIDs resolve to Content Addressed Storage such as IPFS they protect against content drift and the immutability guarantees from CID resolution allows for safely addressing the entire data structure of a given research object. Because a single dPID can address the entire knowledge graph formed by a given Research Object, any linked component becomes granularly and securely citable. As a result, we can import a specific code or data file from a dPID that resolves to a file stored on the IPFS network.

## Next, let's `import.paper()`!

We created two Research Objects with code components stored on the Open State Repository. We committed the first Research Object to the repository and use its decentralized permanent identifier to import specific code files and their associated functions into the code component of the present research object using DeSci Labs open source *desci-fetch* library. This new object was then committed to the repository.

»Hello World!

We call a function from the source research object into the target research object to print Hello World.



Same as above, except we call the functions required to construct a mandelbrot set, which takes inputs from the target research objects and plots it.

## References

- [1] Cited Research Object: <https://dcite.org/57>