Introducing Verite

Verite is an open source verifiable credentials library that makes it easy to issue, present, and verify credentials.[Suggest Edits](#)

We carry credentials with us everyday. Our drivers license or state ID, our library card, our insurance card, and more are all just a few examples of some of the credentials we carry. We present these credentials when requested, and the credentials are verified by the person or entity we are presenting them to. But as the world shifts to an increasingly digital native format, and as people take more ownership over their identity, how can the issuance of, presentation of, and verification of credentials be managed?

The answer is Verifiable Credentials (VCs). VCs are on a web standards track and provide an open data model for digital credentials. We won't go deep into the details of VCs here because the [Verite documentation](#) does a great job of that. Instead, we will focus on what Verite is, why it exists, and how it might be used.

What Is Verite?

Verite is a collaborative and open source initiative spearheaded by the Circle, and its focus is to "provide the governance and standards for the future digital financial ecosystem." Verite is one output of the collaborative effort.

While contributions to the VCs standards themselves is important, and Circle is an active participant alongside the W3C, tools that make it easy to implement and leverage VCs are just as important. As such, Circle began work last year on Verite, an open source library designed to make managing VCs easier.

Currently available in TypeScript and [published through NPM](#) , the Verite library seeks to make it easier to implement VCs in a variety of forms. The library is early and an additional goal of the library is to collect community feedback.

Verite's helper functions simplify VC flows such as requesting credentials, issuing credentials, presenting credentials, and verifying credentials. For the fastest possible start to using Verite, check out the [Quick Start guide](#) in the documentation.

Why Does Verite Exist?

Despite the standards track for VCs, tooling around the implementation of VCs has historically been limited. There have been reference implementations, but a simple developer experience has largely been missing. This is what Verite sets out to solve.

We believe that adoption of VCs is not only about creating good standards. It's also about making the experience of implementation a great one. To this end, Verite has tried to abstract as much of the complexity away as possible. Our documentation dives deep into what's happening behind the scenes, but the Verite library itself feels simple. And that's the goal.

In a sense, Verite is also the canary in the coal mine. It's a test. An experiment. As mentioned above, the secondary goal of Verite's library is to collect community feedback on its implementation. Since the library's release, we've been fortunate enough to receive a number of issues and pull requests in the [Github repository](#) . We encourage the community to continue provide feedback.

How Can Verite Be Used?

We have a [number of demos available](#) in the Github repository that can serve both as reference points and as inspiration. Verite can be used for just about any VC need. In our demos, you'll find:

- Basic Credential Issuance
- Basic Credential Verification
- KYC Compliance and Revocation
- Ethereum-based dApp With Credential Requirement
- Lending Market dApp With KYC Requirement
- Centralized App With Travel Rule Requirement

By design, these demos offer both a mobile credential wallet experience or a credential simulation experience. Verite has no plans to dictate the way credentials are stored and presented. The Verite library can be included in existing cryptocurrency wallets, can be used to built a new digitial credentials wallet, can be used to build mobile apps, and more.

We expect to see some interesting use cases come up in the months ahead because Verite is designed with flexibility in mind. Between community collaboration and that flexibility, implementing and experimenting with VCs will become increasingly easier. We're excited for the future, and we hope you'll help us build that future. Updated5 months ago * [Table of Contents](#) * * [What Is Verite?](#) * * [Why Does Verite Exist?](#) * * [How Can Verite Be Used?](#)