

# PROPOSED: Trust DID Web (did:tdw) DID Method Task Force

- [Background](#)
- [Objectives](#)
- [Context](#)
- [Membership and Joining](#)
- [Deliverables](#)
- [GitHub Repository](#)
- [Intellectual Property Rights \(Copyright, Patent, Source Code\)](#)
- [Milestones](#)
- [Dependencies](#)
- [Meeting Schedule and Notes](#)
- [Mailing List and Communications](#)
- [FAQ](#)

This page is the charter of the proposed Trust DID Web (did:tdw) DID Method Task Force (TDWTF). The [Meeting Page](#) will have links to the meeting agenda and notes if and when the Task Force is approved.

## Background

The "Trust DID Web" (or "Trusted Web, if you same it fast) is a web-based DID method that improves (or extends) `did:web` by providing a verifiable history of authorized (optionally with multiple keys) DIDDoc updates, portability, and (optional) key pre-rotation support in a small, simple package with minimal dependencies. "did:tdw" has what we think are the important features of an easy to use DID Method. DID Controllers (owners) can easily publish the same DIDDoc content as a `did:web` and `did:tdw` together to enable an easy transition -- the "did:tdw" JSON Lines log file "did.jsonl" resides beside the `did:web` "did.json" file. We believe that "did:tdw", with its simplicity, and adequate "ledger-type" features (without a ledger), combined with the publishing simplicity of "did:web," has the potential to be the ultimate DID Method.

DID portability is based on the inclusion of a self-certifying identifier (SCID) -- a GUID that is a required component of the DID string that is derived from the DID's initial DIDDoc. A DID with the same SCID can be moved -- history and all -- to a new web location.

did:tdw includes two DID Core-compliant **services** for handling DID URL paths in the "expected" way for a web-based DID Method. Their compliance with the DID Core specification means that the techniques that can also be used with other DID Methods -- although the simplicity of the semantics might be lost. Notably:

- By default, any `<did>/path/to/file` maps to an HTTPS GET of `<https did path>/path/to/file`
- The DID URL `<did>/whois` uses the "Linked-VP" specification from DIF to resolve a Verifiable Presentation (if published) where the embedded Verifiable Credentials **MUST** have the DID as the subject, and be signed by the DID.

The DID Method implementation dependencies are minimal, with the most "complex" being the use of JSON Canonicalization Scheme (JCS) and JCS EDDSA Data Integrity proofs (links below).

We think we have a lot of solid ideas in the specification and its implementations (Typescript and Python). The goal of this TDW Task Force is to evolve the specification in a working group to welcome new ideas, and to cover open questions, such whether the logging basis of "did:tdw" can be used with other DID Methods, can/should there be a "standard" way for did:tdw DIDs to be also published to a ledger for long term availability, and so on.

### Background Links:

- did:tdw Specification (rendered): <https://bcgov.github.io/trustdidweb/>
- did:tdw Specification (repository): <https://github.com/bcgov/trustdidweb>
- Presentation from IIW 38 (April 2024). The details start at slide 11: [https://docs.google.com/presentation/d/1PHo16asyceRiNKN7UkV8BSmtWtN6Wp3A6\\_9MV0IQ2jg/edit?usp=sharing](https://docs.google.com/presentation/d/1PHo16asyceRiNKN7UkV8BSmtWtN6Wp3A6_9MV0IQ2jg/edit?usp=sharing)
- Typescript Implementation: <https://github.com/bcgov/trustdidweb-ts>
- Python Implementation: <https://github.com/bcgov/trustdidweb-py>
- Linked-VP Specification: <https://identity.foundation/linked-vp/>
- JSON Canonicalization Scheme: <https://datatracker.ietf.org/doc/html/rfc8785>
- eddsa-jcs-2022: <https://www.w3.org/TR/vc-di-eddsa/#eddsa-jcs-2022>

## Objectives

The primary objective of this Task Force is to incubate the `did:tdw` DID Method Specification as a ToIP deliverable. The purpose of this deliverable to define an easy to use, better than `did:web`, web-based DID method, and to possibly submit the specification to an appropriate standards developing organization (SDO),

## Context

did:web is becoming a ubiquitous because it is so simple to use and easy to publish – without a ledger. However, did:web also lacks features that are necessary for a core part of a trusted ecosystems, and are important for DIDs – history, authorized updates, portability. did:web also lacks features that are important for security, such as the ability to commit to a next keypair, without exposing that key. did:tdw adds those crucial features in a simple way, one that is easily understand, easy to implement, and with minimal dependencies. While still susceptible to the mutability of web publishing, did:tdw provides DID portability for planned web location moves, and can be combined with other techniques (registries, archival tools, even the Internet Wayback Machine) to add durability to the DID method.

## Leadership

The proposed leads of the TDWTF are:

1. [John Jordan](#)
2. [Stephen Curran](#)

## Membership and Joining

Prior to participating in the meetings, please ensure that you are a member of the Trust Over IP Foundation (Contributor Membership is free to both organizations and individuals). More details can be found at [this link](#).

To join this TF, add your name to this list:

- [Stephen Curran](#)
- [John Jordan](#)
- [Drummond Reed](#)
- [Eric Scouten](#)
- [Brian Richter](#)

## Deliverables

1. The [did:tdw DID Method Specification](#)
2. The did:tdw Implementor's Guide (currently part of the specification)

There are currently two implementations of the specification available in [TypeScript](#) and [Python](#). To be determined is where those (and other) implementations will be hosted.

## GitHub Repository

The GitHub Repository for the specification is currently housed [here](#), within the BCGov GitHub organization. If the TDWTF is approved, the specification repository will be transferred to the ToIP GitHub organization.

## Intellectual Property Rights (Copyright, Patent, Source Code)

As a Task Force (TF) of the Technology Stack WG (TSWG), the TDWTF inherits the IPR terms from the TSWG [JDF Charter](#).

- Copyright mode: **OWFa 1.0** (available at <https://www.openwebfoundation.org/the-agreements/the-owf-1-0-agreements-granted-claims/owfa-1-0>)
- Patent mode: **OWFa 1.0** (available at <https://www.openwebfoundation.org/the-agreements/the-owf-1-0-agreements-granted-claims/owfa-1-0>)
- Source code: **Apache 2.0** (available at <http://www.apache.org/licenses/LICENSE-2.0.html>)

## Milestones

Key milestones will include, but are not limited to:

1. Publication of the first Draft Deliverable via a GitHub repo.
2. Publication of the final Draft Deliverable.
3. Approval of the Draft Deliverable as a Working Group Approved Deliverable.

The work of this Task Force will be complete when the Working Group Approved Deliverable is approved by the TSWG.

## Dependencies

- None

## Meeting Schedule and Notes

The TDWTF meeting scheduled will be established if this Task Force proposal is accepted. Please see the [ToIP Calendar](#) for the exact meeting times and Zoom links.

See the [Meeting Page](#) for links to the meeting agenda and notes for each meeting (including the Zoom links for joining a meeting and for listening to a recording of the meeting).

## Mailing List and Communications

This task force uses the following for communications. These will be setup if this Task Force proposal is accepted.

- **Slack:** #tswg-trust-did-web <== *This channel encouraged for regular comms.*
- **Mailing List:** There is no separate mailing list for this task force. Use the main TSWG mailing list: [technical-stack-wg@lists.trustoverip.org](mailto:technical-stack-wg@lists.trustoverip.org).

## FAQ

1. **Q:** Why not just use [did:webs](#)?
  - a. The goals of the two DID Methods are quite similar (a better web-based DID Method than did:web), but the approaches are fundamentally different. [did:webs](#) is a KERI-based system that produces as an indirect by-product, a DIDDoc that can be published on the web. Both the use of KERI, and the "indirection" of using KERI states to produce a DIDDoc makes (in the opinion of the Task Force leadership) the DID Method perhaps challenging to implement and maintain if what you primarily need is a DID and DID document. On the other hand, did:tdw is based around the DIDDoc, and the creation and updating of the DIDDoc are the state events that that are logged. The DID Controller defines each version of the DIDDoc they want to publish, and the did:tdw registration process creates the "event" that transitions the DIDDoc from the previous state to the desired, authorized state such that the entire history can be verified by a resolver.