

did:orb

Troy Ronda (SecureKey)

<https://trustbloc.github.io/did-method-orb>

Feb 17, 2021

Background

Orb's design originates from prior experiences with Sidetree and Hyperledger Fabric.

Sidetree:

- Self certifying – DID suffix cryptographically bound to initial state.
- Self controlled – Ordered updates to a DID document form their own verifiable chain from inception to the current state of the DID document.
- VDR based on content-addressed storage and immutable files.
- Batch-based structures assist with performance and storage.

Hyperledger Fabric:

- Endorsement model and batch-based propagation model.

Motivation – Enable Open Federation

- Not be coupled to a particular blockchain or DLT.
 - Decouple propagation coordination into a ledger agnostic protocol.
 - Decouple transaction graph into a ledger agnostic CAS-based structure.
 - Remove the need to choose a common public blockchain and DLT lock-ins.
 - Still allow for ledger usage as a monitorable log.
- Allow for an open federation and replication model.
 - Enable protocols that allow VDRs to interconnect and replicate.
 - Allow a DID to use different servers (and backing ledgers) across updates.
 - Minimize trust in the network and servers.

Motivation – Enable both Web and DHTs

- Content-addressed objects need a mechanism to discover hosts.
 - DHTs are beneficial but a particular network isn't always acceptable.
 - Allow for both Web and DHT models within the same method.
- Enable Web-based discovery:
 - `did:orb:webcas:example.com:bafkr...:EiDy...`
 - Based on WebFinger + REST API.
- Enable DHT-based discovery:
 - `did:orb:ipfs:...:bafkr...:EiDy...`
 - Pluggable model for DHT networks.

Motivation – Enable portability

- VDR objects are replicated across Orb Servers.
 - Can be included in new transactions across servers.
 - Form a graph based on immutable CAS CIDs.
 - Graph can be discovered using CID in the DID string.
- DID controllers can write operations across Orb Servers.
 - Specify the origin that has knowledge of their latest DID operations.
 - Enabled to change origin over time.
 - Canonical DID updated when origin changes (graph CID is updated).

Motivation – Enable monitorable ledgers

- Decouple witness ledgers from the critical path.
 - Allow for Trust but Verify model.
- Leverage the Certificate Transparency model
 - Witnesses observe VDR objects and promise to include in their ledgers.
 - Provide a signed timestamp and a maximum merge delay.
 - Enable monitoring to ensure witnesses follow their promises.
- Use trusted Witness (and origin) timings to resolve late publishing.
- Use origin to enable observers to know if they have the latest operations.

Motivation - Leverage specifications

- DID core compliant.
- Sidetree Protocol to encode DID operations and batches.
- Verifiable Credential format to encode anchors - AnchorCredential.
- JSON-LD Proofs from witnesses form a VC proof chain.
- Certificate Transparency extended for VCs – VCT.
- ActivityPub for propagation.
- WebFinger for Web-based discovery.
- IPFS CIDs and encodings.

The method uses the following ABNF [\[RFC5234\]](#) format:

```
did-orb-format      = "did:orb:" cas-discovery-scheme [":" min-graph-cid]
                      ":" did-suffix [":" long-form-suffix-data]
cas-discovery-scheme = dht-scheme / web-scheme / local-scheme
dht-scheme           = ( "ipfs" )
web-scheme           = "web:" reg-name
local-scheme         = "local"
reg-name             = 1*idchar                ; more constrained than [RFC3986]
min-graph-cid        = 1*idchar
did-suffix           = 1*idchar
long-form-suffix-data = 1*idchar                ; only applicable in the local-scheme
```

See [\[RFC3986\]](#) for the original definition of reg-name and [\[DID-CORE\]](#) for the definition of idchar. [\[SIDETREE\]](#) provides additional explanation for the *did-suffix* and *long-form-suffix-data* elements.

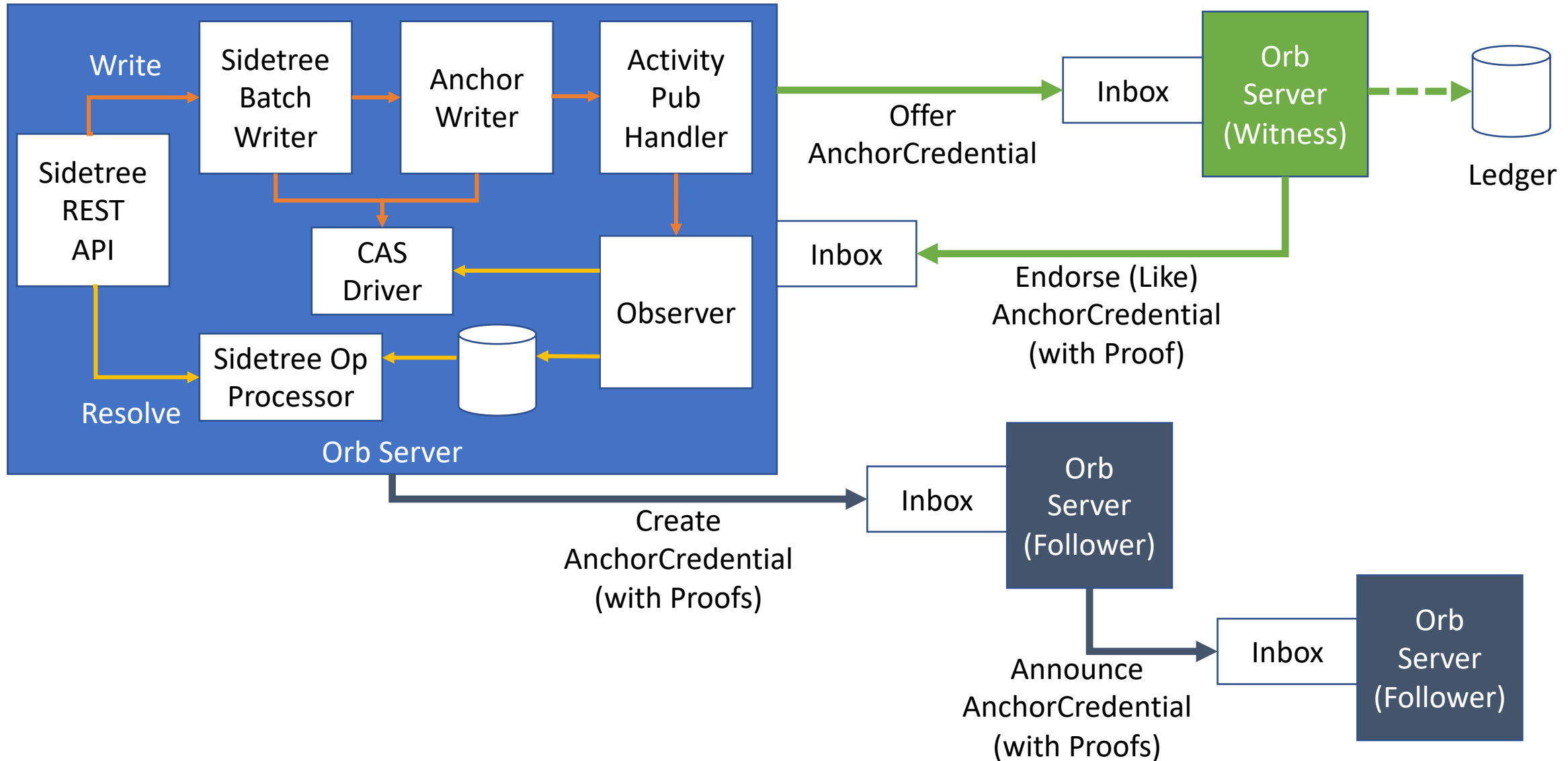
EXAMPLE 1: An Orb DID that uses the Web scheme for content discovery

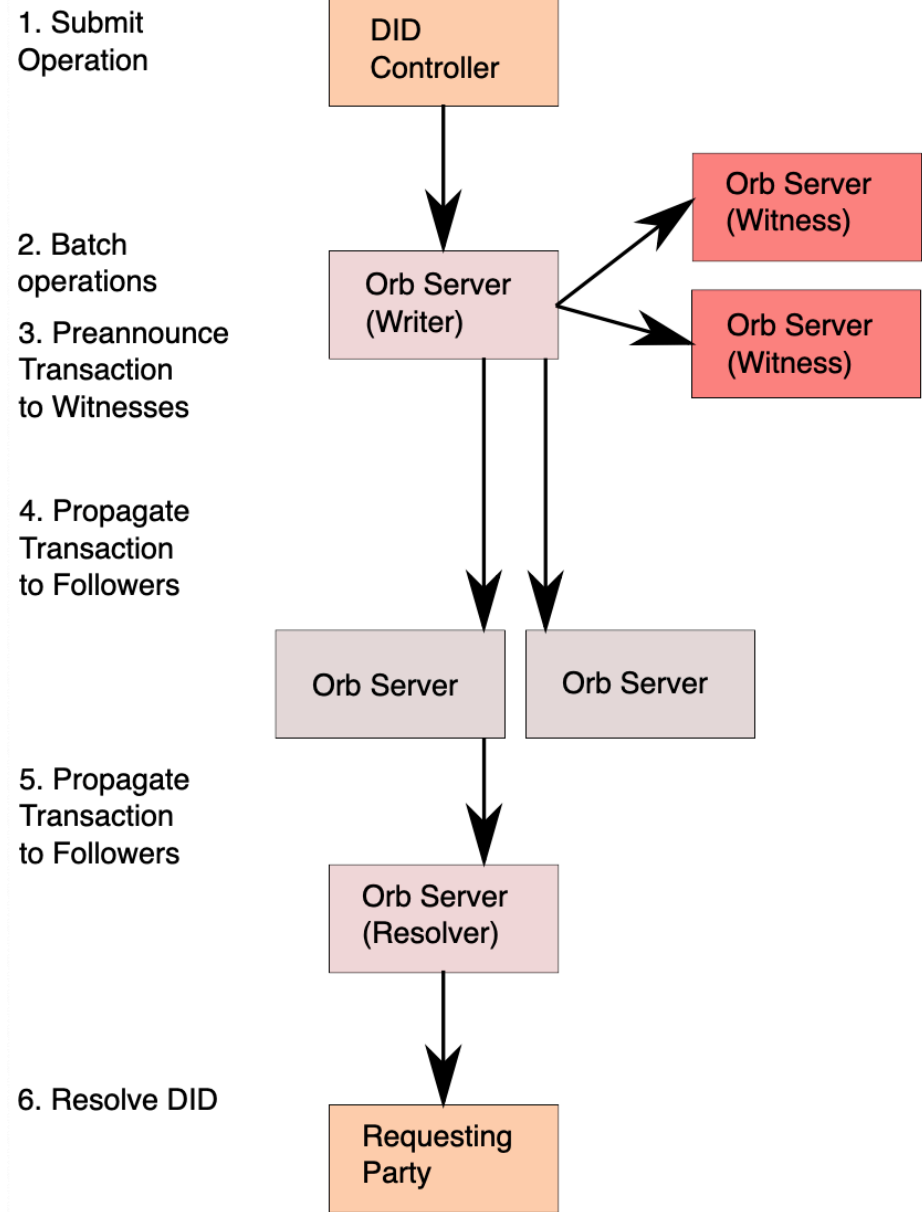
```
did:orb:web:example.com:bafkreiatkubvbkdidscmqynkyls3iqawdqvthi7e6mbky2amuw3inxsi3y:EiDy0QbbZAa3aiRzeCkV7L0x3SERjjH93EXoIM3UoN4oWg
```

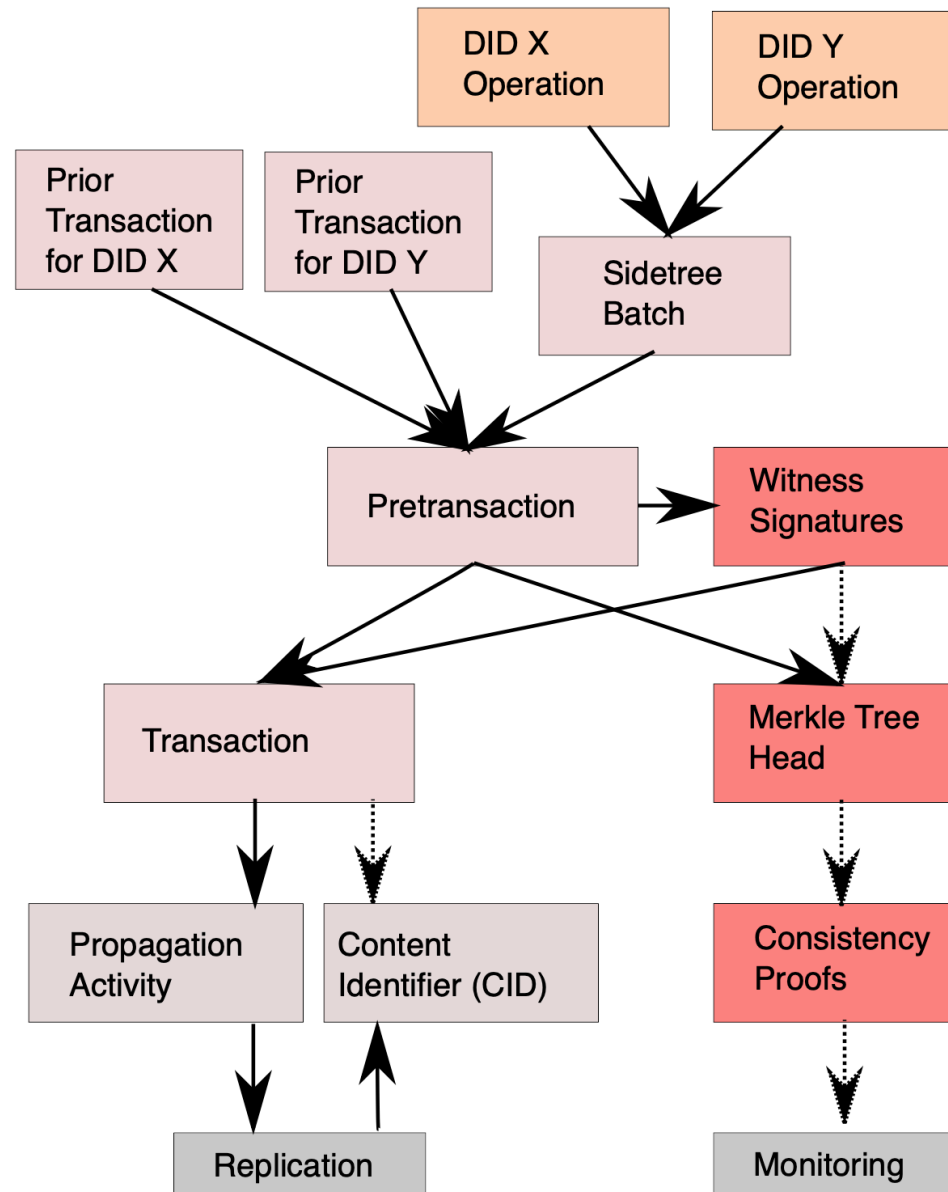
EXAMPLE 2: An Orb DID that uses the IPFS scheme for content discovery

```
did:orb:ipfs:bafkreiatkubvbkdidscmqynkyls3iqawdqvthi7e6mbky2amuw3inxsi3y:EiDy0QbbZAa3aiRzeCkV7L0x3SERjjH93EXoIM3UoN4oWg
```


Orb Network Topology







Late Publishing

