

Passive Identifiers

- Overview
 - What is a Passive identifier?
 - Common types of passive identifier
- Technical requirements
- Shared documents and links

Overview

Data capture requires the provision of fields that are generated deterministically in order to capture and store data inputs. Data capture items are identified by *passive identifiers*.

State	Governing	Non-governing
Authentic	Active	Passive controlled by active
Deterministic	Active controlled by passive <i>(This state cannot exist)</i>	Passive

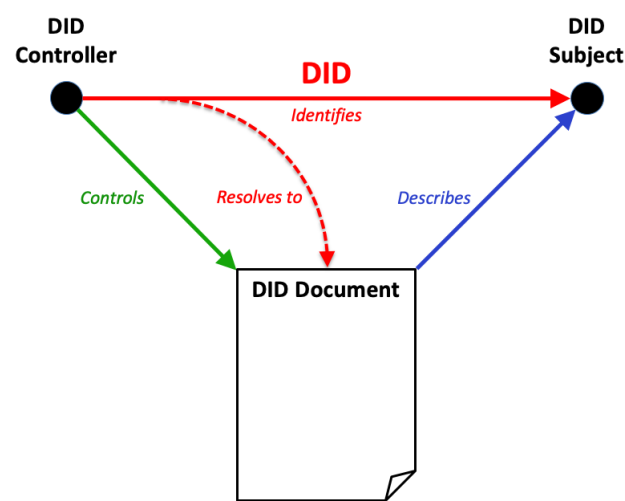
Table: A hash grid table describing the different states of *active and passive identifiers*. The states of *passive identifiers* are highlighted.

What is a Passive identifier?

An identifier that is generated deterministically to identify a non-governing entity, an inanimate object, or a static data item. A passive identifier can either be (1) controlled by an *active identifier* or (2) not controlled.

- **Controlled passive identifier** (*p/a-linkage*)

In general, passive identifiers for non-human resources do not need to be controlled. However, as soon as you have any need to express ownership, control, or management of that resource, you need to relate it to an active identifier. An *active identifier* identifies an entity that has the capacity to govern.



The figure shows an instance when the DID subject may not be the same as the DID controller. In this scenario, the subject may represent a passive non-governing entity, an inanimate object or a static data item with the controller's DIDs being active, requiring a signing key for identity authentication. Through the authentication process, the controller is able to express ownership, control, or management of the passive resource.

- **Uncontrolled passive identifier**

A passive identifier that does not require any form of ownership, control, or management.

Common types of passive identifier

Passive identifier types include:

- **Self-addressing identifier**

An identifier that is deterministically generated from and embedded in the content it identifies, making it and its data mutually tamper-evident.

- **Linking identifier** (*p/p-linkage*)

An identifier that has an association with a cryptographic hash of digital content that can be used for linking information across a number of objects, applications, and/or systems. A linking identifier can be referenced in multiple locations and, as such, changing any single reference of the linking identifier should propagate throughout the daisy chain. As a linking identifier tends to be governed, this threading property works well for data revocation where a governing entity has the capability of revoking all data associated with the linking identifier.

Technical requirements

Passive identifiers are assigned to the [Semantic domain](#). Technical requirements include:

- [Cryptographic hashes](#). A cryptographic hash is a checksum or digital fingerprint derived by performing a one-way hash function (a mathematical operation) on the data comprising a computer program (or other digital files). Any change in just one byte of the data comprising the computer program will change the hash value. A hash value is, therefore, a unique fingerprint for any program or other digital files.

Shared documents and links

- [Active and Passive Identifiers](#)