A basic conceptual model for verifiable identity

l. boldrin

luca.boldrin@infocert.it
luca.boldrin@unipd.it

Introduction

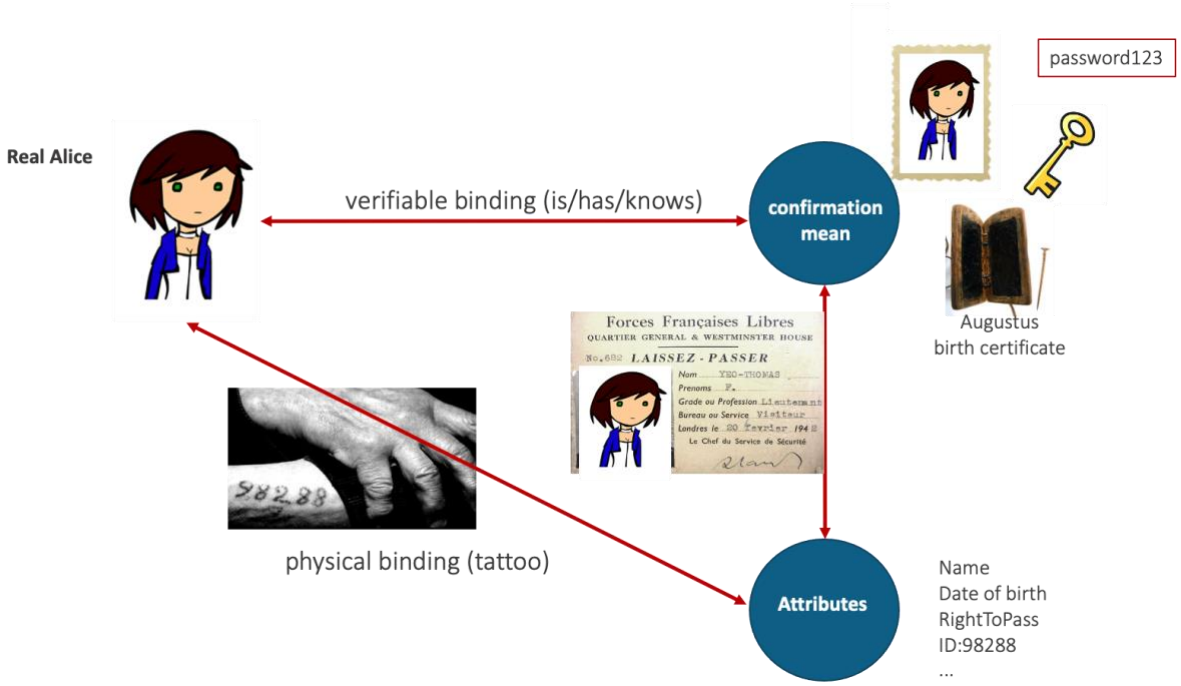
We are used to think of a "digital identity" as a set of attributes associated to an individual. Individuals can voluntarily disclose some of these attributes to third parties who need to know them in order to engage in some transactions. The tricky point is how to make these attributes "verifiable", so that the relying party can conveniently get convinced about their truth. This is the core identity problem: "how to associate a human being with a set of attributes in a verifiable way", which dates back to the pre-digital world. The problem is even trickier in the digital realm.

This memo is a contribution to the community discussion on the **modeling of a verifiable digital identity**. It aims at an abstract and simplified model, explicitly omitting some relevant aspects (like the use of presentations, selective disclosure, holder vs. subject, etc.). These notes were largely inspired by recent discussions on confirmation methods[1][2] and on the use of identifiers[3].

A naïve recap of pre-digital identity

The core identity problem: "how to associate a human being with a set of attributes in a verifiable way", has been addressed in various ways in the course of the pre-digital human history[4], but has always been problematic. Given the need to bind the flash-and-bones person to some data, a tattoo was probably one of the strongest, tamper-resistant "verifiable binding".

[1] https://lists.w3.org/Archives/Public/public-credentials/2023Feb/0000.html
[2] https://docs.google.com/presentation/d/1-uPVyl3S-vPvy4HqL6BcjN0xTu9AvqxFfwowqwzcXpo/edit#slide=id.g1f24e2c0aad_14_10
[3] https://www.kuppingercole.com/sessions/5064/2
[4] See e.g. https://www.business-reporter.co.uk/technology/the-history-of-digital-identity



More convenient (and less invasive) ways of addressing the issue required the introduction of an intermediate object (the "confirmation mean"). The first part of the binding (real Alice ←→ confirmation mean) was performed through **something the person is/have/knows**. The second part (confirmation mean ← → attributes) was performed by some physical "document". Roman emperor Augustus (27 BC–14 AD) is credited to have introduced **birth certificates** [5] (wooden diptych with waxed surfaces) in 4 AD. They provided a verifiable binding by "possession" of the document. In this case the certificate provides both the confirmation mean and the binding to the attributes (name, citizenship...).

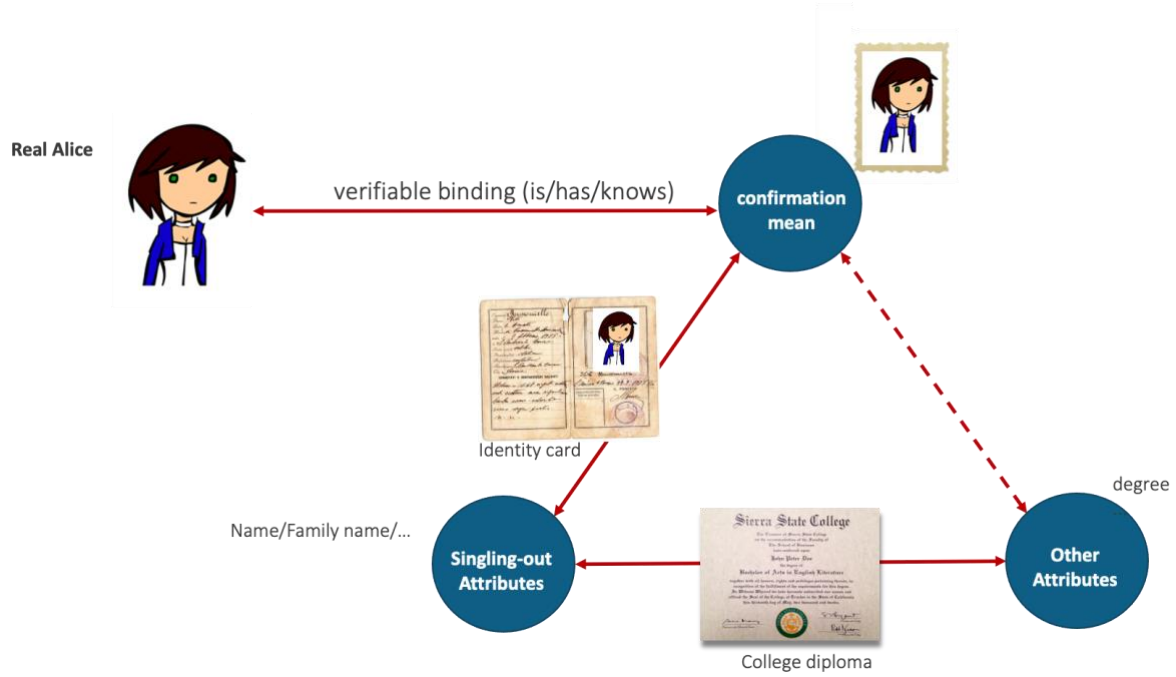
In most cases the confirmation mean does not conveys itself the binding. This is especially true when the confirmation mean is a biometric representation of the person (a picture, a fingerprint). In the figure above the confirmation mean (Alice's picture) is distinct from the binding (the laissez-passer). The same confirmation mean can be used to construct several bindings.

In the physical realm, in modern times, the binding is often achieved with a physical document (e.g., an identity card binding the picture with name/family name, a laissez-passer, a driving licence, etc.).

As it happens, attributes have different "singling-out power"[6]. The bindings related to attributes with the strongest singling-out power have a special role, especially when they bind to a "uniquely singling-out" attribute. Leveraging on these strong bindings, it is possible to create additional "attribute← → attribute" bindings.

[5] https://en.wikipedia.org/wiki/Birth_registration_in_ancient_Rome

[6] I prefer "singling-out attribute" to "identifying attribute" since the second is often associated to the real-world identity of the person (name, family name). "singling-out" only aims at unicity.



In the physical realm, a recurrent singling-out attribute is the concatenation `name||familyName||dateOfBirth||placeOfBirth`. In the picture above

- an identity card is the verifiable binding between the picture and the singling-out attribute `name||familyName||dateOfBirth||placeOfBirth`
- a diploma certificate is the binding between the singling-out attribute and the degree.

Of course, it still makes sense to have direct binding between the confirmation mean (the photo) and non-identifying attributes. This is the common practice for the driving license, the company badge, etc. (which, incidentally, regularly also convey some identifying attributes like the name).

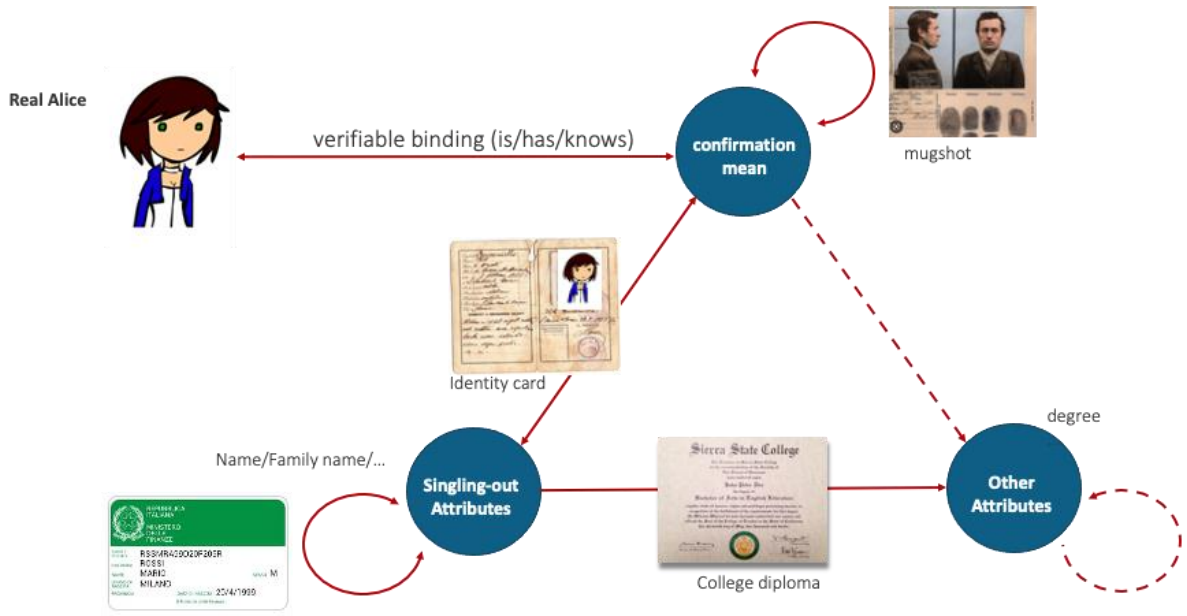
Additionally, bindings can also happen between attributes and between confirmation means:

- A fiscalCode document associates `name||familyName||dateOfBirth||placeOfBirth` to a (unique) code, forming a binding between two singling-out attributes.
- A police mugshot including a picture and a fingerprint provides a binding between two confirmation means.

Potentially, there might be bindings between non-singling-out attributes,

- a claim from the Italian Ministry of Education stating that students having grade A in UK are recognized grade 9 in Italy
- a claim from an Healthcare institution asserting that people with pathology A are given priority code X in case of access to Emergency Room.

These bindings normally come in the form of a public piece of regulation.



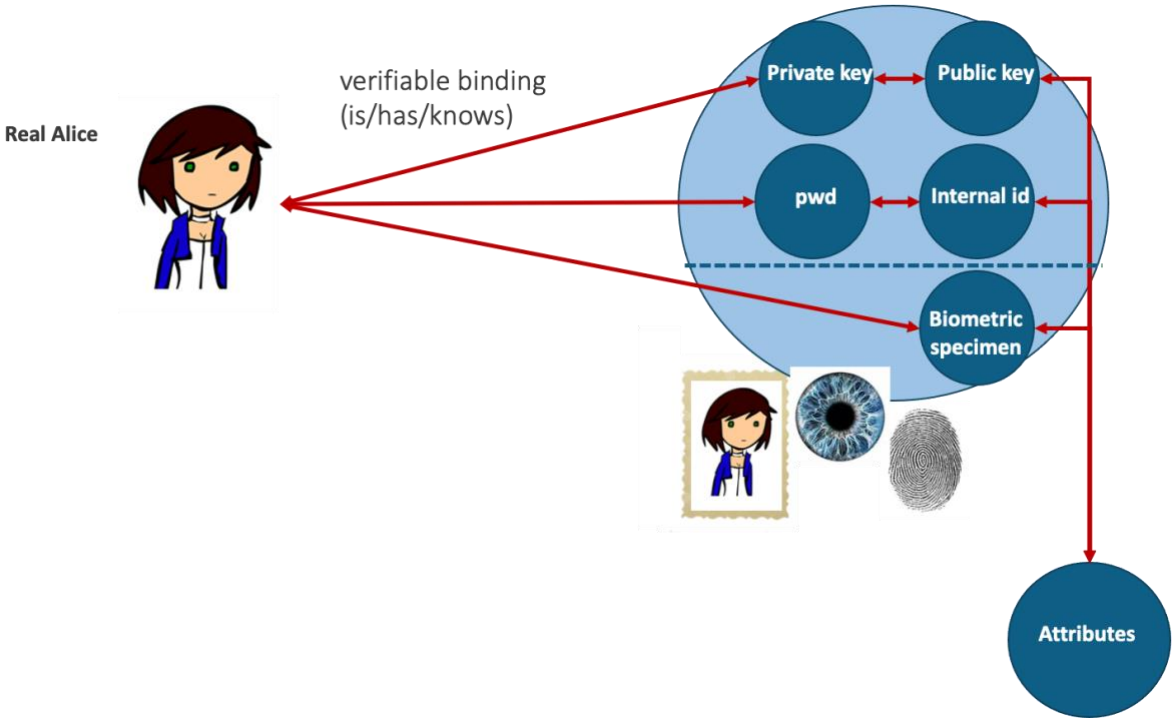
It is worth noting that **bindings involving non-singling-out attributes are one-way**: the binding (`fiscal_code:RSSMRA99D20F205R, degree:computer_science`) can not be reversed, nor the binding `(pathology:A, emergencyPriority:X)`

(uni-directional arrows in the picture above).

Identity in the digital realm

In the digital setting, some adjustments to the basic model are required.

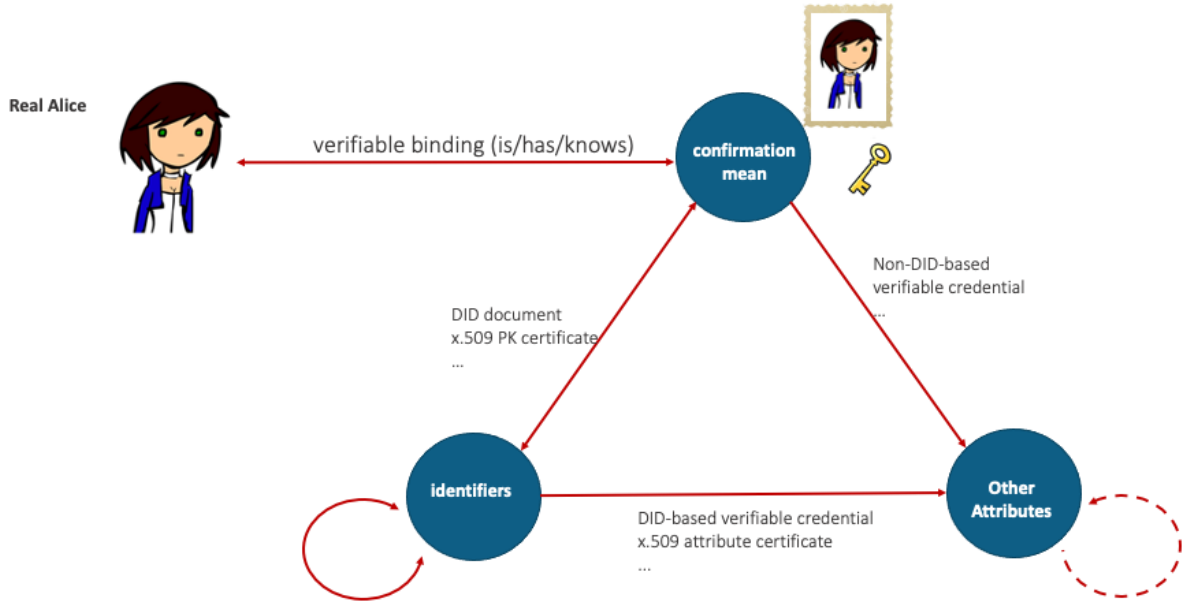
To start with, some of the traditional confirmation means do not adequately perform in remote settings, which is a frequent situation in the digital realm.



Confirmation means, in the digital case, mostly belong to one of two categories:

- A **biometric specimen** (alike the physical realm). In this case the bind to the attributes is a digital document including the specimen and the claimed attributes (e.g., a mDL document)
- A **secret**. This might be
    - a private key, in which case there is a cryptographic binding to the public key. The public key is then used as the anchor for binding to attributes (public key certificates, DID documents, etc.)
    - a password, in which case there is a binding to an internal id managed by some operator. The internal id is the anchor toward additional attributes
    - something else...

As for the physical realm, it is conceivable (though not necessary) that there are some “privileged attributes”, with high singling out power (ideally, with one-to-one mapping to the person). We stick with the usual practice to refer to these singling-out attributes as “**identifiers**”.



Obviously, nothing prevents a direct binding between a confirmation mean and a generic attribute, without going through an identifier. The major reason for using an identifier is to "isolate" the confirmation mean, which may change over time, from other attributes. If the confirmation mean needs changing, only its binding to the identifier would be affected (e.g., key rotation).

Again, there might be bindings between identifiers, between confirmation means and, possibly, between non-singling-out attributes as well.

Note that in order to bind an identifier to another attribute, **we must be sure of the unicity of the identifier** – this can be achieved in many different ways, including the use of a distributed ledger which forces the unicity of the identifier, by using a proprietary namespace, by creating a large enough random number, etc.

Binding attributes in the digital realm

A binding is, in its essence, a way to convince the verifier that the two attributes are linked or, more precisely, a convincing proof that **whoever can prove to be associated to the first attribute, can also prove (one-way) to be associated to the second one**.

While in the physical realm the binding mostly occurs by engraving the two attributes on a physical substrate, convincing a verifier in the digital realm may happen in many ways, e.g.:

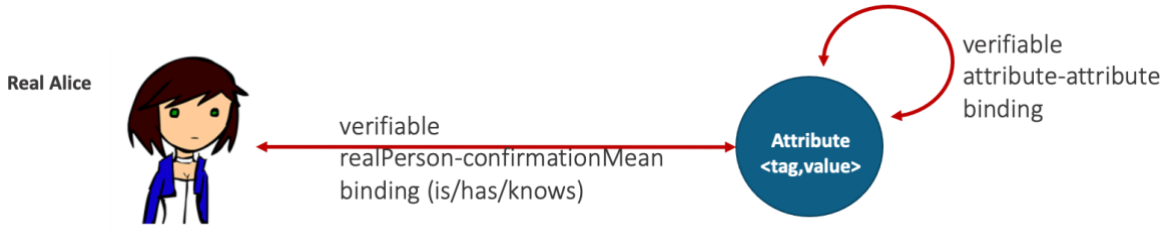
- A trusted entity vouches for the binding by digitally signing a file
- A trusted entity vouches for the binding by providing the association through a digital service (e.g., a REST API) on a secure channel
- The association is engraved in a DLT
- The binding can be calculated by an algorithm
- Any other proof that the verifier is willing to accept ...

We mostly concentrate on the first form of binding, which is the approach taken in X.509 attribute certificates (binding a public key certificate to whatever attribute), W3C verifiable credentials in the

DID-based flavour, SAML tokens, etc.. Nevertheless, any form of binding is relevant as far as a verifier is willing to accept it.

A basic model

While, for practical reasons, it makes sense to distinguish between different kinds of attributes (confirmation means, identifiers, other attributes), we can generally think of a simplified model like the one below.



In fact, we do not need to deal separately with different types of attributes. We only distinguish between attribute-attribute binding and person-attribute binding (which, normally, is a binding to a confirmation mean). The characterization of attributes can be performed via a tag. A possible model for an attribute can be in the form of an ordered couple:

$$a=\langle tag, value \rangle$$

where `tag` belongs to a space of possible attribute names, while `value` belongs to the space of the respective values for the attributes. The `tag` provides the semantics of the attribute, and may help the verifier to decide whether to treat it as a confirmation mean, an identifier, etc. As a matter of fact, there is need for a standardized ontology of tags to establish a shared semantics between issuers and verifiers.

As for the attribute-attribute bindings, independently of what they bind, they can be naturally represented as a couple:

$$\langle a_1,\ a_2 \rangle$$

where a_1, a_2 are the two attributes to be bound. In most situations the binding will be vouched for by some validation information (a "proof"), e.g. a digital signature on the couple of attributes provided by some trusted entity or any other information which helps verifying it.

Again, note that **the association is not reversible**, i.e. $\langle a_1,\ a_2 \rangle$ is different from $\langle a_2,\ a_1 \rangle$.

How to verify Alice's attributes

Based on the above model the problem "how to associate Alice with a set of attributes in a verifiable way", boils down for a verifier to:

1. get one or more confirmation means (a picture from a scanner in the airport, a public key provided by Alice herself...)
2. verify the binding between Alice and the confirmation mean(s) according to the specific modality for the confirmation mean (is/has/knows)
3. get a set of bindings of which at least one starts from the confirmation mean (directly from Alice or any other sources)
4. verify each binding using the respective validation information

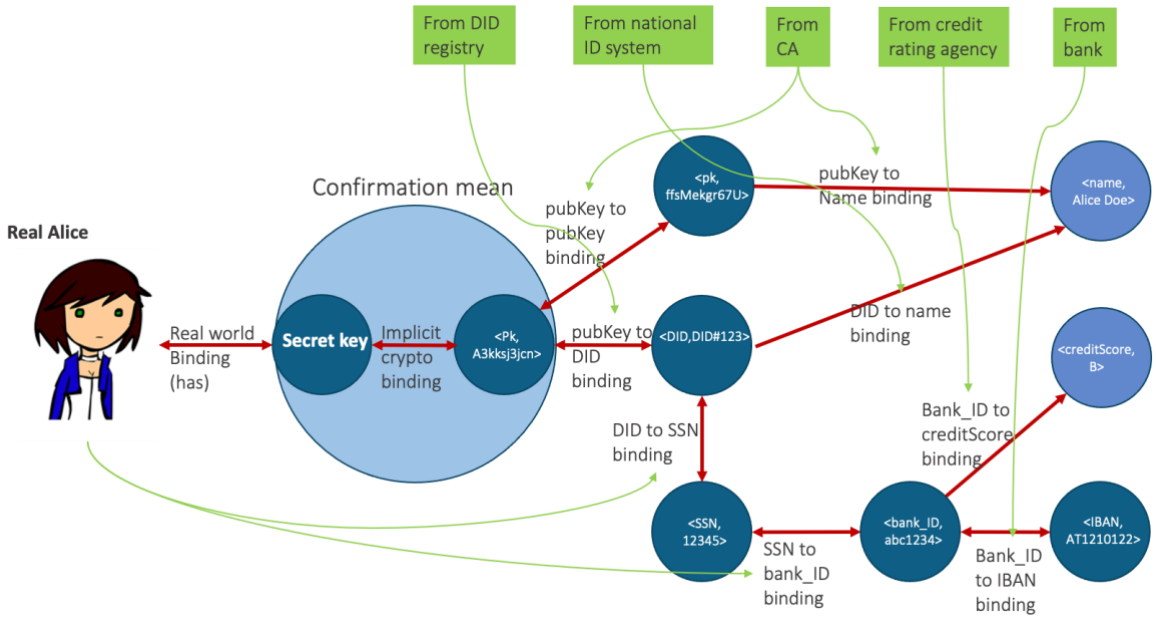
5. follow the chain of bindings starting from a confirmation mean to the desired attributes.

Note that, while this is the normal approach in the self-sovereign paradigm, **there is no need for bindings to come directly from Alice**. We can conceive scenarios were some or all bindings come from different sources (which would qualify as "holders"). In fact, as far as the verifier is concerned, **the source of bindings is irrelevant**, as long as they are verifiable, i.e. there is a proof for them which can convince the verifier.

As a special case, it may happen that the verifier is NOT interested in ascertaining that the interacting person is indeed Alice but is only interested in getting some information about Alice. In this case, the interacting person (if any) is just a "holder" as any other source. The verifier will not activate the confirmation mean validation, and will only follow points 3, 4, 5 from the list above.

An example

In the (somehow artificial) picture below, a verifier collects and chains verifiable bindings from several sources in order to get 3 attributes related to Alice who is a remote user of its online services (name, creditScore, IBAN). Bindings may be collected via interaction with Alice's wallet, via access to some ad-hoc repository, by interacting with a specific issuer, ...



Some of the bindings come from Alice, who got them via a previous interaction with a trusted issuer. Additionally, it may make sense for a verifier to **reach the same attribute via different paths** (like for "name" in the picture). It may also make sense to bridge to the real Alice via different confirmation means (multi-factor authentication).

Most of the attributes in the picture are "identifiers", i.e. reasonably unique (dark blue). Only credit score and name are not, so arrows to these attributes is uni-directional.

The bindings in the picture need not all be the “signed credentials” we are used to. Some of them may be verified on the base of a trust relation with the source or by any other mean which the verifier accepts as a proof.

Some remarks

Along this note concepts were simplified, and many relevant items were omitted, including:

- **Selective disclosure:** it was not considered, on the assumption that issuers can provide deal atomic credentials. **Predicates** were left out from this discussion.
- **Presentations:** they were not considered, assuming that a verifier is only interested in validating some attributes as witnessed by a chain of proofs. Aspects like the intention of Alice to present these attributes is out of scope.
- **Holder vs. subject:** this distinction was not relevant in the model, since the verifier may decide whether to activate the confirmation mean validation or not.
- **Unlinkability** was not addressed, including the specification of a wallet endpoint.
- **Degrees of trust** was not considered, we limited to a black/white approach to trusting bindings.

Appendix - Sketching a formal model

This section is a sort of a formal exercise and can be skipped with no prejudice. It aims at sketching a formal model (syntax and semantic) capturing the above concepts.

Syntax

We introduce a simplified syntax for exemplification. The syntax is completely arbitrary, but can easily be mapped to existing JSON, XML, ASN1... notations and appropriately extended.

The alphabet includes:

- $t_1, \ldots t_n$ - a set of atomic terms for tags
- $v_1, \ldots, v_m$ - a set of atomic terms for values
- ‘b’ ‘(‘ ‘)’ ‘:’ ‘,’ – a set of punctuation symbols

A term for an attribute is a concatenation: $t_i{:}v_j$ (like, for example, `eyeColour:brown`). The set of terms for attributes, is then

$$Att = \{\ t_i{:}v_j \mid \forall\, i \in \{1, \ldots n\}\ ,\ j \in \{1, \ldots m\}\ \}$$

A formula in our language is defined as $b(t_i{:}v_j, t_h{:}v_k)$ where $t_i{:}v_j$, $t_h{:}v_k$ are terms for attributes. The language is defined as the set of formulae, i.e.

$$Lan = \{\ b(t_i{:}v_j, t_h{:}v_k) \mid \forall i,h \in \{1, \ldots n\},\ \forall j,k \in \{1, \ldots m\}\ \}$$

Semantics

A formula $b(t_i{:}v_j, t_h{:}v_k)$ is intended to mean: “whoever can prove to be associated with attribute $t_i{:}v_j$ can also prove to be associated with attribute $t_h{:}v_k$”. We capture this in the semantics.

Let $M = (I, A, \sigma)$ be a model for our language, where

- $I = \{\ i_1, \ldots i_r\ \}$ – intended to represent a set of individuals

- $A = \wp(I)$ – (A is the set of parts of I) is the space of all possible attributes. In our model an attribute coincide with the set of individuals who share that attribute (e.g., the set of individuals with $eyeColour:brown$). Identifiers are special attributes corresponding to unary sets $\{i_j\}$
- $\sigma: Att \rightarrow A$ is a function which maps each term of the language $t_i:v_j$ to an element of A

For any formula $b(t_i:v_j, t_h:v_k) \in Lan$ we define

$$M \models b(t_i:v_j, t_h:v_k) \text{ iff } \sigma(ti:vj) \subseteq \sigma(t_h:v_k)$$

Which reads: the model M satisfies the binding $b(t_i:v_j, t_h:v_k)$ if and only if the interpretation of the first attribute is a subset of the interpretation of the second attribute.

A set of formulae $F \subseteq Lan$ is said to be coherent if there exists a model which satisfies all formulae in F. A set of formulae $F \subseteq Lan$ is said to be incoherent if there is no such model.