Ponderings on a (Self-Sovereign) Identity Framework

Rieks Joosten {rieks.joosten@tno.nl}

Scope

In the business (human) domain, I think we can safely limit ourselves to situations where information (not: data) is being processed in order to reach a business decision (conclusion), and further limit ourselves to decisions that are needed in business transactions. We use the basic models of the Design & Engineering Methodology for Organizations (DEMO) for modeling (business) transactions and related decision making. Our primary focus is on the so-called 'promise' decision, i.e. where a business decides whether or not to engage in a business transaction (i.e. commit to a transaction-agreement proposal). We assume that any business B will commit to a proposed transaction agreement iff (1) the merits for B outweigh the costs for B and (2) the level of risk that B will run is acceptable to B. Please note that an individual human being is also considered to be a business here (and vice versa).

Information Context, arguments and business decisions

We also assume that each business B has its own 'view of the world' or information context as I like to call it. It consists of (1) B's knowledge of entities that exist in the real world, (2) schemes that B uses to sort such entities into classes of similar entities, (3) relations that state which entities of one class relate to which entities of another class in some (subjective) given meaning, and (4) rules that B uses to model applicable constraints on entities and relations. A business logic that B applies is part of the rules of B's information context. A statement that B uses can be modeled as a set of entities (better: entity references) and a reference to the relation (that is in B's information context) that defines their meaning. An argument for a business decision (by B) is a business logic (set of rules) of B and a set of statements in B's information context such that when the rules are applied to the statements, there is a yes/no outcome, which is (the value of) the business decision. Any reasoning and decision making by B takes place within B's information context, which is pretty much the definition of subjectivity.

Committing to a business transaction – validity of the commitment decision

As mentioned before, B will commit to a business transaction with some other business(es) iff (1) the merits of the transaction outweigh its costs and (2) B finds that the (residual) risks are acceptable. In the negotiations, B (and the other parties) negotiate in order to maximize their profits while keeping the risks acceptable AND convincing the other parties to commit as well. Such negotiations may cause B (as well as the others) to revise the business logic that is being used (computing profits differently, or re-assessing the risks), which also changes its information need. While negotiating, it is imperative for B to consider whether or not (1) it (still) has a sufficiently correct understanding of

the statements that the other party provides (semantics), (2) these statements are 'sufficiently true', and (3) the business logic that it uses to come to the commitment decision is still valid. Misunderstanding the meaning of a statement and/or using a false statement and/or using invalid business logic may all invalidate the outcome of its commitment decision.

Electronic actors and Electronic business transactions

IT does not have the flexibility, adaptability etc. that humans have. This implies that electronic actors (IT) are incapable of conducting business transactions in the way that human actors do. The challenge is to determine in what ways electronic actors can optimally support business transactions – reducing the workload for human actors – and find out what constraints must then be satisfied. So where we first had a fully human business-to-business interaction, say $H_1[B_1] – H_2[B_2]$ (where $H_i[B_j]$ means that human H_i negotiates and decides on behalf of business B_j in the interaction), we are now looking at $(H_1,E_1)[B_1] – (H_2,E_2)[B_2]$ (where $(H_i,E_k)[B_j]$ means that there is a human H_i and some electronic actor (running software) E_k that both act on behalf of business B_j in the interaction). The idea is to minimize the work for all H_i and maximize the work that the E_k do. Obviously, B will need to decide what part of the work it offloads to its electronic actors, and what part remains to be done by humans.

An Electronic Acting Prerequisite

Electronic actors behave differently from human actors. Where humans have 'gut feelings' that allow them to decide whether or not the actor they are negotiating with is 'authentic' or 'trustworthy', whether or not statements are true or business logic is valid, electronic actors must rely on implicit decisions (e.g. that any business logic that comes from digital profile is valid), or on arguments for such decisions that have been made explicit (e.g. a statement with meaning X that is attested to by business Y is considered to be true). Instructing electronic actors w.r.t. the kinds of decisions they must make, how to construct the arguments for that and how to decide whether or not the meaning and truth of the statements and the the business logic are all valid, is something that each individual business needs to do for itself. And this is a far from trivial and usually grossly underestimated challenge.

Another Electronic Acting Prerequisite

If an electronic actor E1[B1] wants to engage with another electronic actor $E_2[B_2]$ in order to negotiate and commit to a business transaction – i.e. a transaction between B_1 and B_2 – then E_1 must have some way to determine which business E_2 is actually representing. Note that this is equally true when B_1 and/or B_2 are individual people. This should lead to requirements for the electronic actors themselves: they should be 'well behaved', and other electronic actors must have means/mechanisms to decide whether or not this is sufficiently the case – this decision, like any other, has to be made explicit by the business that they represent, and the argument that leads to this decision may vary depending on the kind of transaction in which this assurance is needed. We currently think that well-designed attestations may be helpful here.

Form Filling Assistance

In our current thinking, the (semi)electronic negotiation of business transactions and the corresponding decision making is well-served by envisaging the electronic actor not only displaying a web-form (that, when filled in, constitute statements to be used in the commitment argument) that a human end-user can fill in, but also by accompanying that form with further meta-data for its fields that will enable an app of the end-user to assist in filling in the form, and provide further assurances (e.g. verifiability of the claims) that may be required according to the meta-data. In the more distant future, the electronic actor may also accompany the form with the business logic that it will use to assess the information, so that this app will be in a position to determine the optimal way to fill in the form.

Verifiable Claim (VC) Issuance

Whenever two businesses (typically an end-user and a 'regular' business) have conducted an (electronic) business transaction, this will necessarily have expanded both their information contexts. This also means that both businesses can issue VCs to one another, attesting to (the meaning and truth of) existing and/or newly gained statements as a result/product of the transaction. Such VCs would (functionally) be stored in or by the app that the business uses to assist in the form filling described above. This does not preclude that one business may request VCs from another business in a separate transaction, which – by recursion – might require the filling in of a form…

Technical Infrastructure

A technical infrastructure should help ALL businesses that want to take on this challenge. This implies that it should be agnostic to specific businesses (instances) while supporting the generic ways of work. We envisage a protocol layer (message exchange) that would nicely fit with the W3C VC draft standard, that will accommodate RDF(S), perhaps OWL, SHACL, … This needs more thought though.

A technical infrastructure should be able to sit on Sovrin/Indy, but also work with Attribute Based Credentials (ABCs, such as IRMA), and other technologies (e.g. X.509 certificates, BlockCerts, …)

A technical infrastructure should be wrapped such that it can be added on to existing products with a minimum amount of hassle (e.g. there should be a WordPress plug-in).