JYVÄSKYLÄN YLIOPISTO
UNIVERSITY OF JYVÄSKYLÄ

Blockchain Governance Compass University of Jyväskylä, Finland

Gabriella Laatikainen,

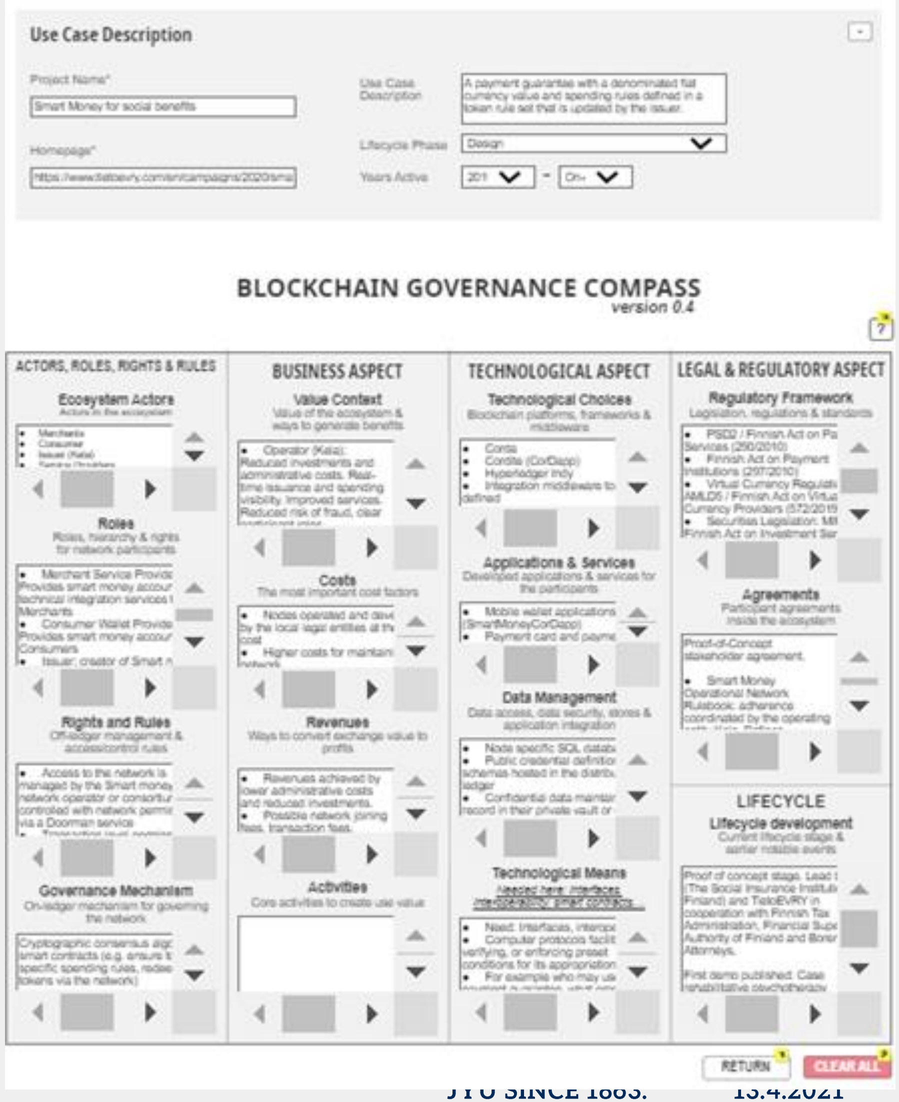
Taija Kolehmainen, Juha-Pekka Tolvanen, Pekka Abrahamsson

TrustOverIP Ecosystem Foundry Working Group Meeting

25.3.2021

Early thoughts to develop a tool for collaboration and brainstorming based on theory…

- The first idea was to create a web application similar to the Business Model Canvas, to model blockchain ecosystems at a bird eye view
- The tool could help ecosystem orchestrators to design and operate their ecosystems built around decentralized ledger technologies
- Then we found Domain-Specific Modeling (DSM) more suitable for this task as DSM enables code generation and automatization; it also allows the use of domain-specific concepts
- The language is currently in the first evaluation phase, the second phase of the development is planned
  - The tool was evaluated by 37 SSI ecosystems designed by the students of a course at the University of Jyväskylä
  - We modeled a couple of real-world ecosystems



Use Case Description

Project Name*: Smart Money for social benefits

Homepage*: https://www.tietoevry.com/en/campaigns/2020/sma

Use Case Description: A payment guarantee with a denominated fiat currency value and spending rules defined in a token rule set that is updated by the issuer.

Lifecycle Phase: Design

Years Active: 201 - On-

BLOCKCHAIN GOVERNANCE COMPASS
version 0.4

ACTORS, ROLES, RIGHTS & RULES

Ecosystem Actors
Actors in the ecosystem

- Merchants
- Consumer
- Issuer (Kela)
- Service Providers

Roles
Roles, hierarchy & rights for network participants

- Merchant Service Provide
Provides smart money accour
technical integration services t
Merchants
- Consumer Wallet Provide
Provides smart money accour
Consumers
- Issuer; creator of Smart n

Rights and Rules
Off-ledger management & access/control rules

- Access to the network is
managed by the Smart money
network operator or consortiu
controlled with network permi
via a Doorman service

Governance Mechanism
On-ledger mechanism for governing the network

Cryptographic consensus algo
smart contracts (e.g. ensure t
specific spending rules, redee
tokens via the network)

BUSINESS ASPECT

Value Context
Value of the ecosystem & ways to generate benefits

- Operator (Kela):
Reduced investments and
administrative costs. Real-
time issuance and spending
visibility. Improved services.
Reduced risk of fraud, clear

Costs
The most important cost factors

- Nodes operated and deve
by the local legal entities at th
cost
- Higher costs for maintain
network

Revenues
Ways to convert exchange value to profits

- Revenues achieved by
lower administrative costs
and reduced investments.
- Possible network joining
fees, transaction fees.

Activities
Core activities to create use value

TECHNOLOGICAL ASPECT

Technological Choices
Blockchain platforms, frameworks & middleware

- Corda
- Cordite (CorDapp)
- Hyperledger Indy
- Integration middleware to
defined

Applications & Services
Developed applications & services for the participants

- Mobile wallet applications
(SmartMoneyCorDapp)
- Payment card and payme

Data Management
Data access, data security, stores & application integration

- Node specific SQL datab
- Public credential definitio
schemas hosted in the distrib
ledger
- Confidential data mainta
record in their private vault or

Technological Means
Needed here: interfaces, interoperability, smart contracts…

- Need: Interfaces, interope
- Computer protocols facilit
verifying, or enforcing preset
conditions for its appropriation
- For example who may us

LEGAL & REGULATORY ASPECT

Regulatory Framework
Legislation, regulations & standards

- PSD2 / Finnish Act on Pa
Services (290/2010)
- Finnish Act on Payment
Institutions (297/2010)
- Virtual Currency Regulat
AMLD5 / Finnish Act on Virtua
Currency Providers (572/2019
- Securities Legislation: MI
Finnish Act on Investment Ser

Agreements
Participant agreements inside the ecosystem

Proof-of-Concept
stakeholder agreement.

- Smart Money
Operational Network
Rulebook: adherence
coordinated by the operating

LIFECYCLE

Lifecycle development
Current lifecycle stage & earlier notable events

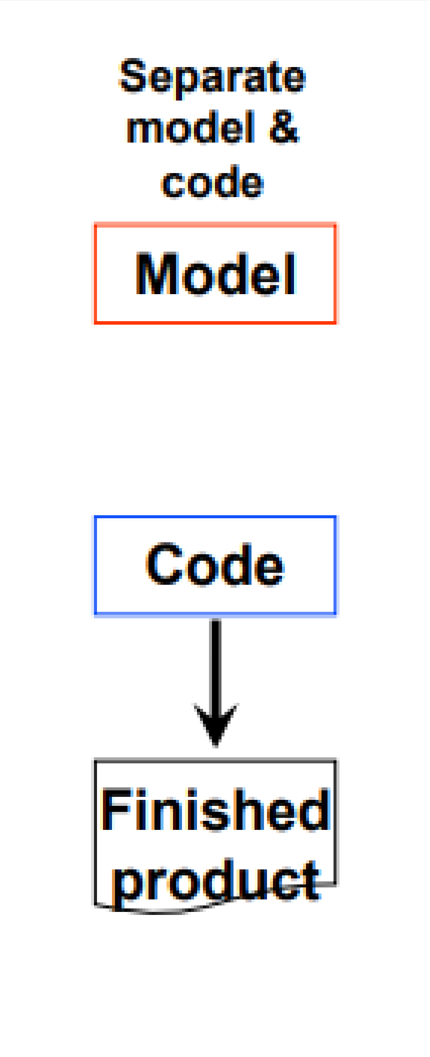
Proof of concept stage. Lead t
(The Social Insurance Institut
Finland) and TietoEVRY in
cooperation with Finnish Tax
Administration, Financial Supe
Authority of Finland and Borer
Attorneys.

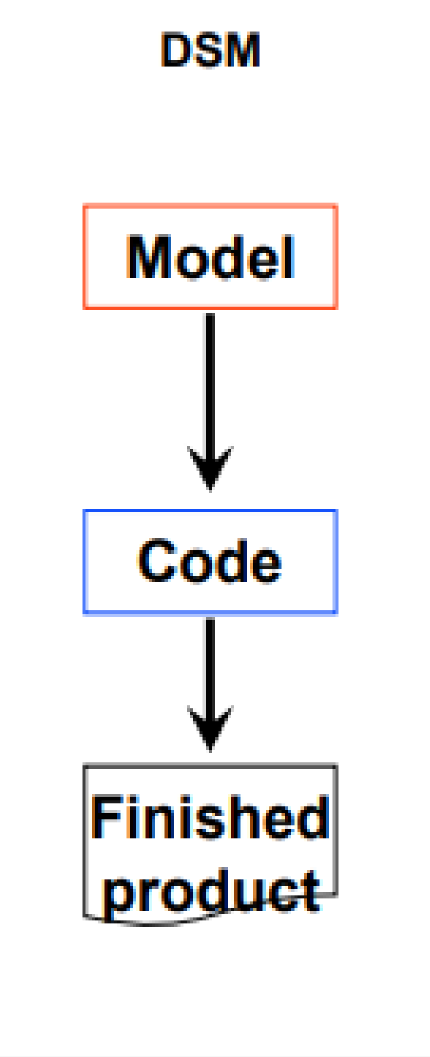
First demo published. Case
rehabilitative psychotherapy

Domain-Specific Modeling (By Juha-Pekka Tolvanen)

Modeling in system development

- Complex
  - many levels of change, overwhelming amount of detail, different views
- Uncertain
  - why to change, what to change, how to change
- Contextual and contingent
  - past history, development group, domain, technology





Why domain-specific modeling? (By Juha-Pekka Tolvanen)

- Captures domain knowledge (as opposed to code)
  - Raise abstraction from implementation world
  - Uses domain abstractions
  - Applies domain concepts and rules as modeling constructs
  - Narrow down the design space
- Leverages in-house expertise to build automation
- Lets developers design products using domain terms
  - Apply familiar terminology
  - Solve the RIGHT problems
  - Solve problems only ONCE! (directly in models, not again by writing code, round-trip etc.)

"The entire history of software engineering is that of the rise in levels of abstraction"

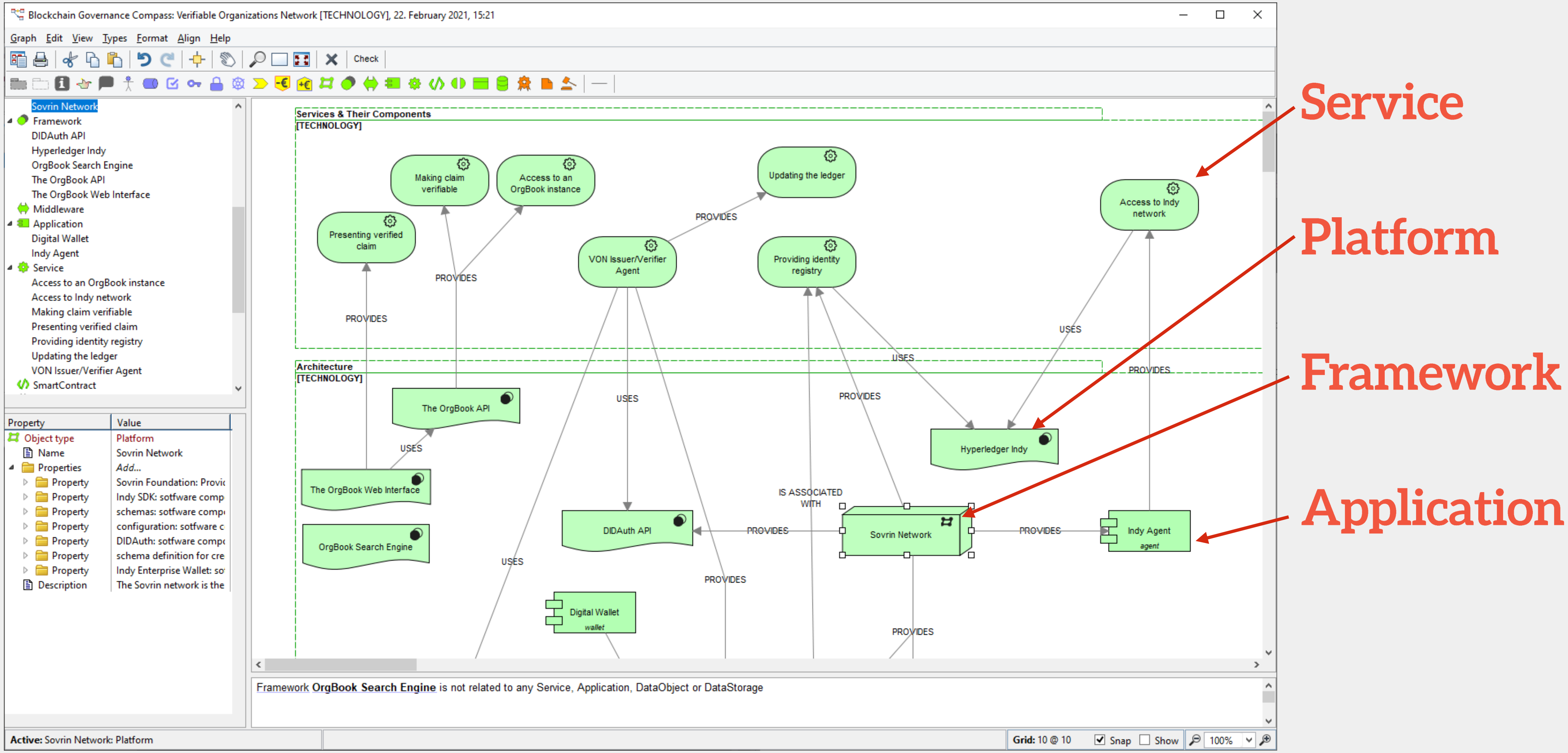
Defining a DSM solution: Steps (By Juha-Pekka Tolvanen)

1. Identify abstractions
   - Concepts and how they work together
2. Specify the metamodel (abstract syntax)
   - Language concepts and their rules
3. Create the notation (concrete syntax)
   - Representation of models
4. Define the generators (semantics)
   - Various outputs and analysis of the models

The process is iterative:

- Define part of language, model with it, define more...

Blockchain Governance Compass: example



Blockchain Governance Compass: Verifiable Organizations Network [TECHNOLOGY], 22. February 2021, 15:21

Sovrin Network
- Framework
  - DIDAuth API
  - Hyperledger Indy
  - OrgBook Search Engine
  - The OrgBook API
  - The OrgBook Web Interface
- Middleware
- Application
  - Digital Wallet
  - Indy Agent
- Service
  - Access to an OrgBook instance
  - Access to Indy network
  - Making claim verifiable
  - Presenting verified claim
  - Providing identity registry
  - Updating the ledger
  - VON Issuer/Verifier Agent
- SmartContract

Property	Value
Object type	Platform
Name	Sovrin Network
Properties	*Add...*
Property	Sovrin Foundation: Provic
Property	Indy SDK: sotfware comp
Property	schemas: sotfware comp
Property	configuration: sotfware c
Property	DIDAuth: sotfware comp
Property	schema definition for cre
Property	Indy Enterprise Wallet: so
Description	The Sovrin network is the

Framework **OrgBook Search Engine** is not related to any Service, Application, DataObject or DataStorage

Active: Sovrin Network: Platform

Grid: 10 @ 10 ☑ Snap □ Show 100%

Service

Platform

Framework

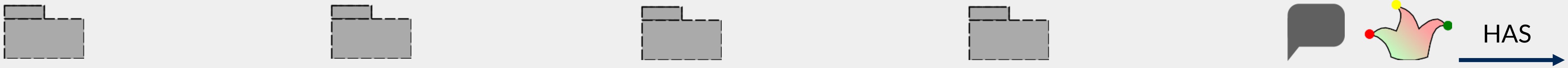
Application

Developing the BC Governance Compass Language

- The objective was to build a domain-specific language for modeling ecosystems built around decentralized ledger technologies
  - A model should have the following characteristics: abstract, understandable, accurate, predictive and inexpensive.
  - Provides a set of domain concepts to discuss the system under development
  - Automatic rules to assure that some relevant aspects are defined (e.g. mandatory data, unique data, compulsory relationships)
- Implemented in MetaEdit+, a tool for DSM, available at https://metacase.com/

[1] Selic, B. (2003). The pragmatics of model-driven development. IEEE software, 20(5), 19-25.

BG Compass Language Building Blocks: General Elements



Governance Container

- Actors and roles
- Rights, rules and responsibilities
- Incentives

Business Container

- Business activities
- Financial aspects

Technology Container

- Architecture
- Services and their components
- Data

Legal & Regulatory Context Container

- Laws, acts and regulations
- Standards
- Agreements

General Elements

- Comment
- Joker
- Relationship (is, has, uses, provides, applies to, complies with, is associated with)

BG Compass Language Building Blocks: Governance

Icon	Name	Description	Examples
🧍	Actor	An entity who is capable of performing behaviors or activities in the ecosystem.	Individual, business, organization, community or authority, etc.
▭	Role	A characteristic set of behaviors or activities undertaken by ecosystem actors.	Non-monetary could be e.g., reputation, visibility, shared norms, gaining experience, networking, collaborating, etc.
☸	Incentive	Motivational factors of the actors or roles to take actions. Type: Monetary/Non-monetary	Verifier, issuer, holder, developer, etc.
🔑	Right	A right or privilege to perform a certain behavior or activity.	Access, development, voting, decision, ownership, control rights, etc.
🔒	Rule	A regulation or principle that governs conduct in the ecosystem.	Development or governance rules, etc.
☑	Responsibility	A behavior or action that actors or roles can be held accountable for.	E.g., maintain a node.

BG Compass Language Building Blocks: Business

Icon	Name	Description	Examples
➤	Business Activity	A collection of business behavior that an actor or role can perform to create and capture value.	Individual, business, organization, community or authority, etc.
-€	Cost Factor	A financial representation of a cost incurred due to an asset, resource, activity or service necessary for value creation or capture. Type could be fixed/variable.	Hardware costs, cloud costs, coordination cost, migration cost, training costs, sales costs, search costs, negotiation costs, etc.
+€	Revenue Model	Could be e.g., marketing, node hosting, providing specific services, development, sales, production/manufacturing, accounting, logistics, etc. ich an actor or a role captures value. Additional properties: • Base (cost, value, competition), • Temporal rights (one time payment, monthly or annual fee, pay per use), • Type (service fee, advertising, micro-payments, other)	Usage fee, subscription, licensing, advertising, micropayments, etc.

BG Compass Language Building Blocks: Technology

Icon	Name	Description	Examples
▱	Platform	In this context, a platform refers to technological architecture (tools, libraries, and reusable components) that facilitate value creation.	Corda, Ethereum Enterprise, Hyperledger Indy, etc.
●	Framework	Software libraries (components, interfaces and tools) that enable the development of a technological solution.	Exonum, Hyperledger Fabric, Openchain, etc.
⇔	Middleware	Software that facilitates the integration of different components within a unified interface.	E.g., Chainlink
🟩	Application	A technological component that is built to encapsulate and perform a functionality.	E.g., mobile phone applications
⚙	Service	An explicitly defined exposed technology behavior. Different network services.	Agency services, value tokenization, etc.

BG Compass Language Building Blocks: Technology

Icon	Name	Description	Examples
</>	Smart Contract	A piece of code that represents a self-executing digital contract. They are automatic rules that are embedded into technology.	Rule validation, identity implementation, credential revocation, credential issuance, etc.
●	Consensus Mechanism	A fault-tolerant method of authenticating and validating a value or transaction on a distributed ledger.	Proof-of-stake, proof-of-work, decided between nodes, etc.
🟩	Data Object	The collection and retention of digital information. Additional property: Type (credential, token, other).	A verifiable credential (VC), public DIDs, e-receipts, etc.
🛢	Data Storage	A data object represents a piece of information structured for automated processing.	Off-ledger (business) databases, public databases, wallet, etc.

BG Compass Language Building Blocks: Legal & Regulatory Context

Icon	Name	Description	Examples
🔨	Law, act, or regulation	A set of rules created and enforced through social or governmental institutions to regulate certain aspects or behavior in the ecosystem. Additional type: Regulation, Law or Act.	Payment institution act, national laws, GDPR, etc.
🏵	Standard	A set of specification, established norm, guideline or requirement that serves to ensure compatibility, safety and quality of certain aspects or components of the ecosystem.	Technical standards, industry standards, etc.
📄	Agreement	Legally enforceable contract among ecosystem actors.	Trade agreements, business contracts among the actors, foundation bylaws, etc.

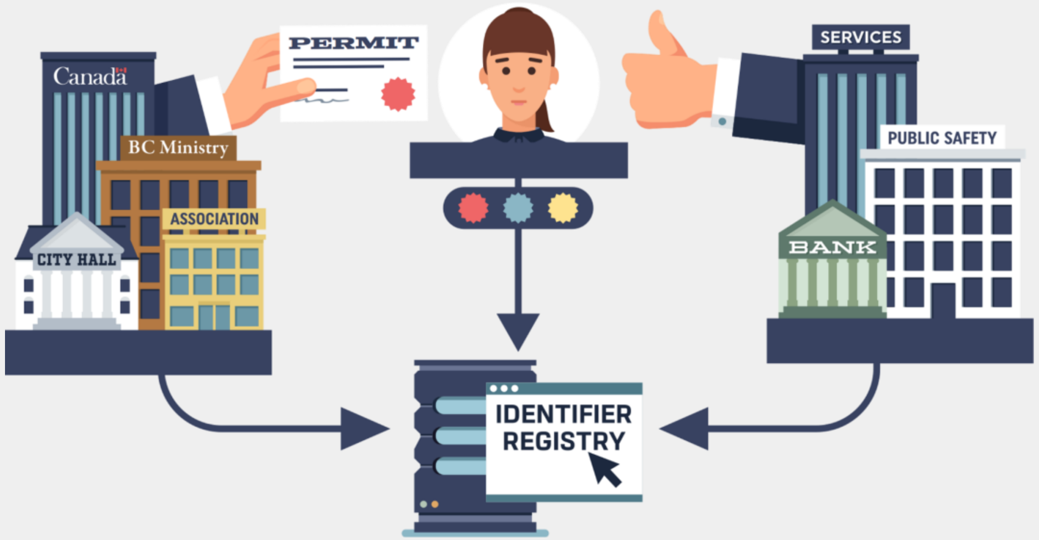
Modeling an Example Ecosystem: Introducing Verifiable Organizations Network

- The goal: To enable organizations to **conduct business online in a trusted manner**.
- The big challenges:
  - a lack of services issuing verifiable credentials about organizations,
  - a lack of organizations with the ability to hold verifiable credentials about themselves.
- The VON project is bootstrapping the trust attribute of the SSI approach for **organizational entities**.

“We aim to create a trusted digital network of verifiable data about organizations, which is globally connected, interoperable, secure, and easy to join.”

https://vonx.io/about/

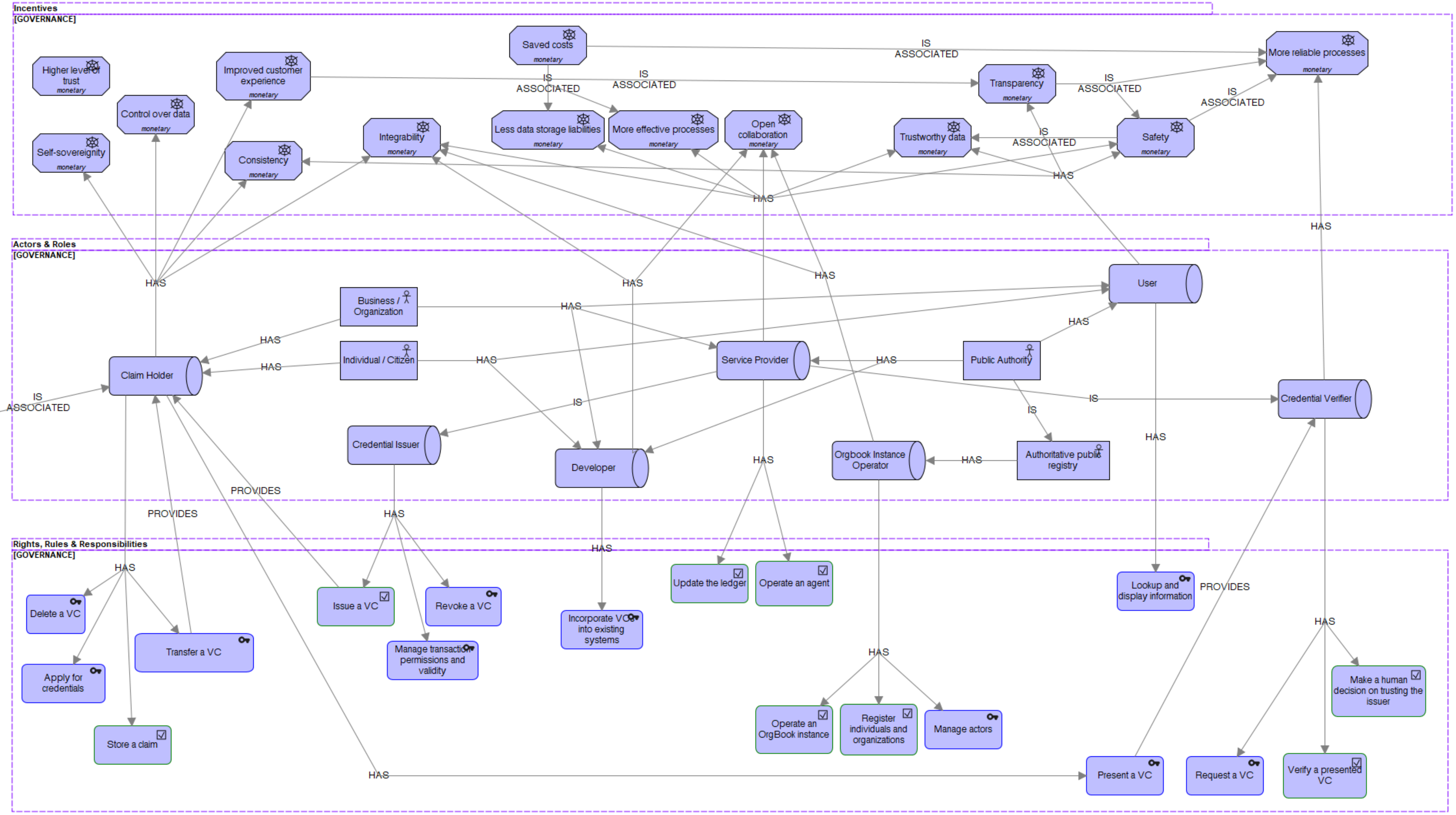
Modeling an Example Ecosystem: Introducing Verifiable Organizations Network



https://vonx.io/getting_started/von-overview/

- Types of VON participants (participating in VON or operating components of VON)
  - **Authoritative Public Registries**
  - **Permits and Licenses Issuers**
  - **Registration, Permit and Licence Verifiers**

- Why would they want to join the project?
  - Users do not need to re-type the information for each service
  - The **information** always comes from a trusted source:
    - Issued by the issuer
    - Issued to OrgBook
    - It has not been forged
    - It has not been revoked



Incentives
[GOVERNANCE]

Saved costs *monetary*
More reliable processes *monetary*
Improved customer experience *monetary*
Higher level of trust *monetary*
Control over data *monetary*
Self-sovereignity *monetary*
Consistency *monetary*
Integrability *monetary*
Less data storage liabilities *monetary*
More effective processes *monetary*
Open collaboration *monetary*
Transparency *monetary*
Trustworthy data *monetary*
Safety *monetary*

Actors & Roles
[GOVERNANCE]

Business / Organization
Individual / Citizen
Claim Holder
Credential Issuer
Developer
Service Provider
Public Authority
User
Orgbook Instance Operator
Authoritative public registry
Credential Verifier

Rights, Rules & Responsibilities
[GOVERNANCE]

Delete a VC
Apply for credentials
Transfer a VC
Store a claim
Issue a VC
Revoke a VC
Manage transaction permissions and validity
Incorporate VCs into existing systems
Update the ledger
Operate an agent
Operate an OrgBook instance
Register individuals and organizations
Manage actors
Lookup and display information
Present a VC
Request a VC
Verify a presented VC
Make a human decision on trusting the issuer

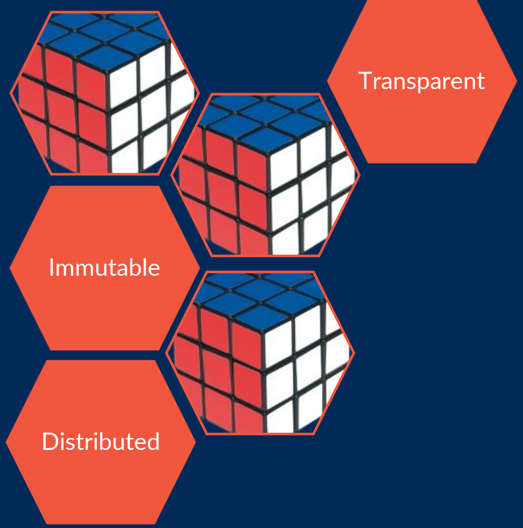
HAS
IS
IS ASSOCIATED
PROVIDES

Future plans

- At the moment, we are evaluating the language by modeling more real world ecosystems - do you have one that could be modeled?

- Currently the language is a planning and collaboration tool -> we would like to develop it further and transform it into a real governance tool
  - Code generators allows us to generate code from the model
  - That is, people without technical knowledge could use this tool to design and automatize governance rules in the ecosystem

Thank you – Please comment and ask questions!



Transparent

Immutable

Distributed

Thank you!

Questions and comments?