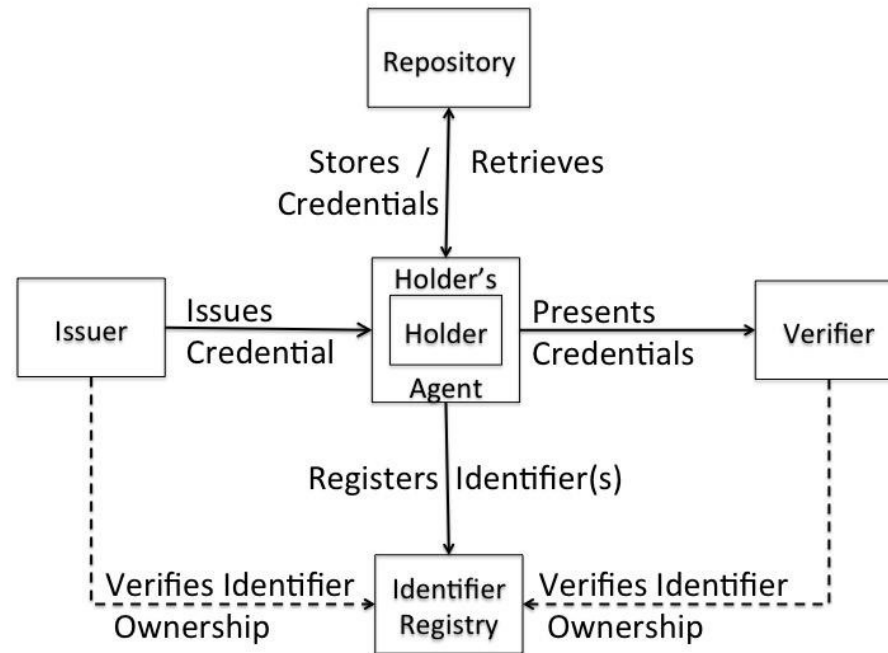


Explorations of Category Theory for Verifiable Credentials

Internet Identity Workshop # 35

Brent Shambaugh

Verifiable Credentials Lifecycle



Decentralized Identifier Architecture

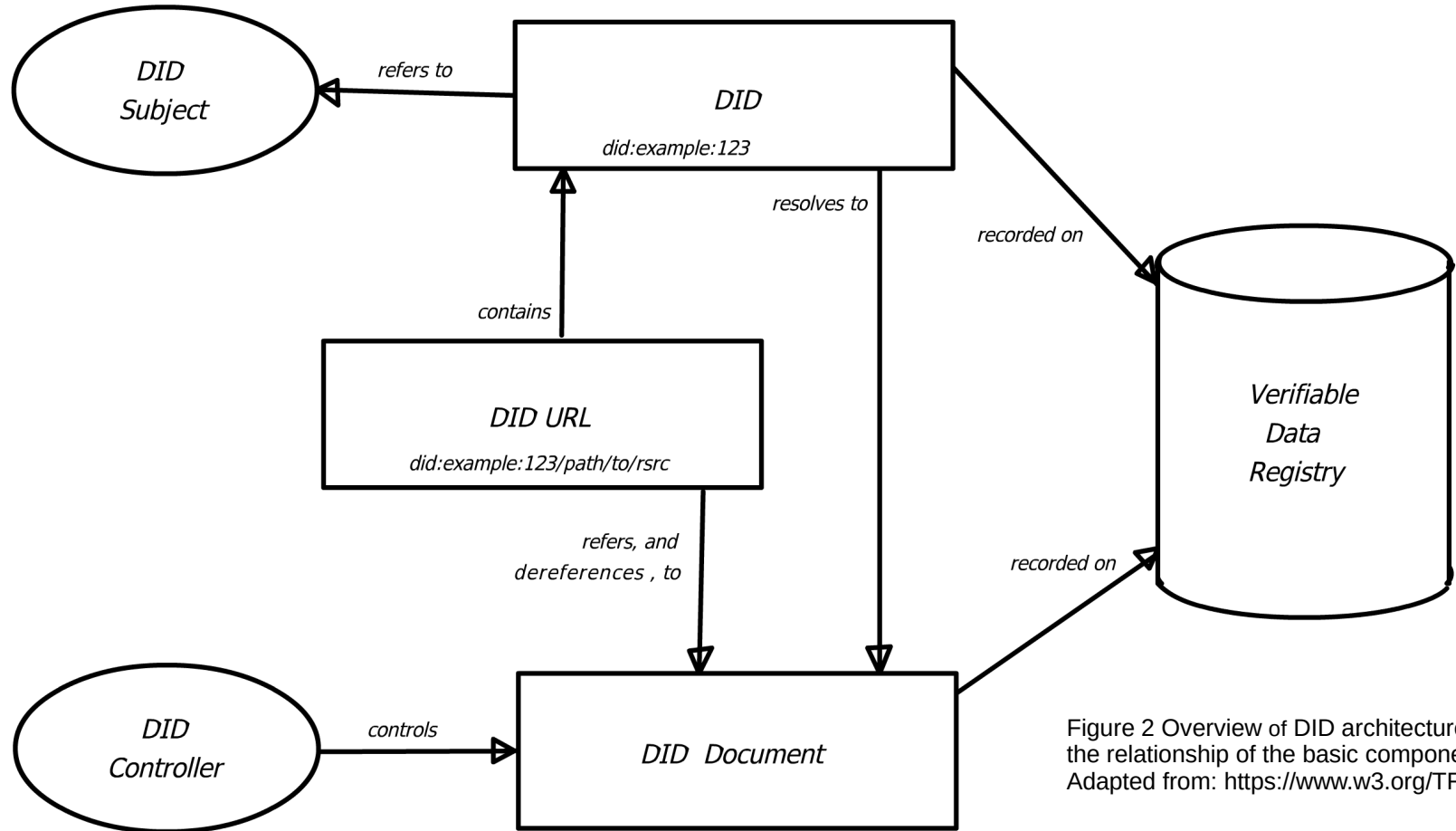
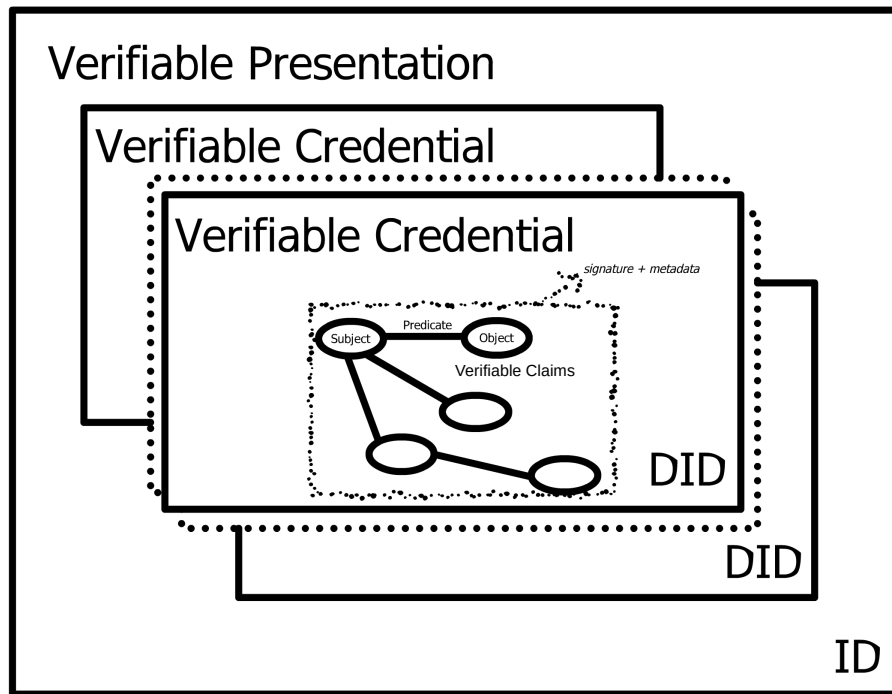
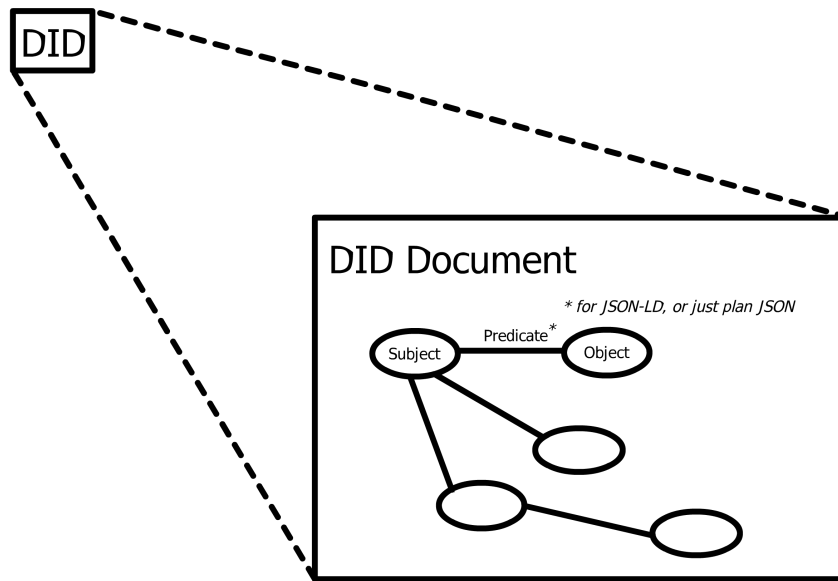


Figure 2 Overview of DID architecture and the relationship of the basic components.
Adapted from: <https://www.w3.org/TR/did-core/>

Data Models



Inspired by:
<https://www.w3.org/TR/vc-data-model/>,
<https://identity.foundation/presentation-exchange/spec/v2.0.0/>



Inspired by:
<https://www.w3.org/TR/did-core/>

Elliptic Curves

$$Ax^3+Bx^2y+Cxy^2+Dy^3+Ex^2+Fxy+Gy^2+Hx+Iy+J=0 \quad \text{General Elliptic Curve}^1$$

$$y^2=x^3+ax+b \quad \text{Weierstrauss Form}$$

{used for secp256k1, secp256r1, secp384r1, secp521r1}^{2,3}

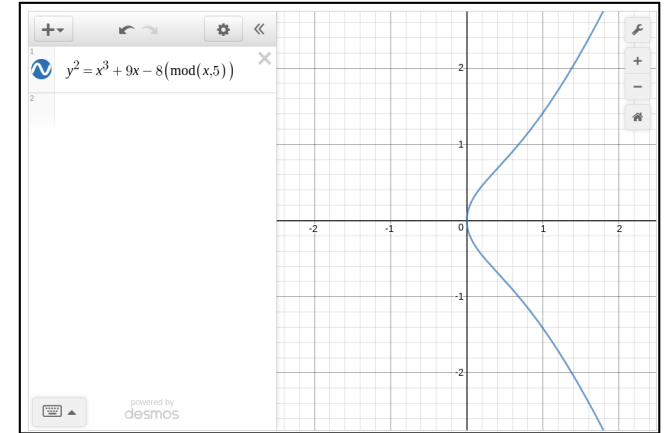
$$y^2=x^3+ax^2+x \quad \text{Montgomery Form {used for ed25519}}^3$$

a and b are large integer constants in sources 2 and 3

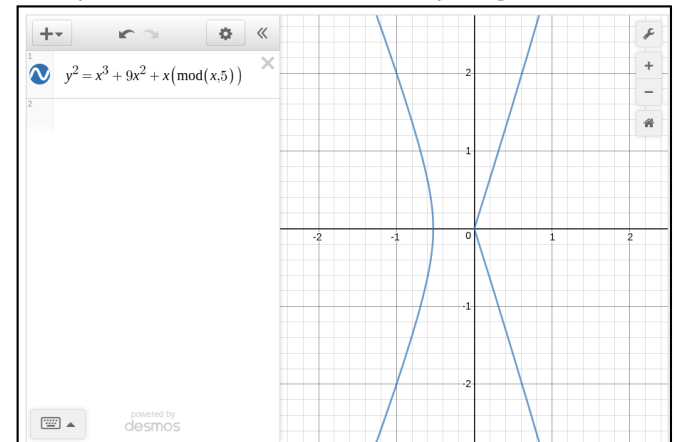
1. <https://mathworld.wolfram.com/EllipticCurve.html>
2. <http://www.secg.org/sec2-v2.pdf> , pg. 9 - 12
3. <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-186-draft.pdf>, pg. 38

Graph Plotter Link:
<https://www.transum.org/Maths/Activity/Graph/Desmos.asp>

Graph Plotter :: An Online Graphing Calculator



Graph Plotter :: An Online Graphing Calculator



Defintion of a Group

A group must have the properties:

Closure: For any **a** and **b**, **$a * b$** is also in the group

Associativity: For any **a,b,c** in a group, **$a * (b * c) = (a * b) * c$**

Identity Element: For any **a** in the group **$a * 1 = a$**

Inverse Element: For any **a** in the group, there is an **a^{-1}** as well, such that **$a * a^{-1} = 1$**

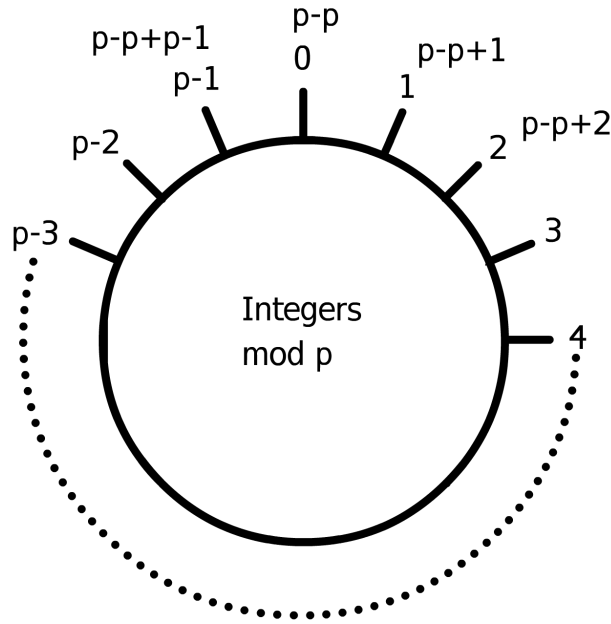
Quoting, page 92, Real World Cryptography, David Wong, Manning Publications

Groups in ECC

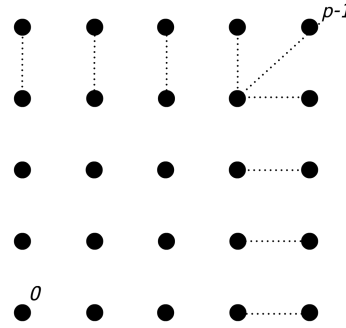
$$y^2 = x^3 + ax + b \pmod{p}$$

F_p p is a large number in a finite integer Field

$$y^2 = x^3 + ax^2 + x \pmod{p}$$



Addition and subtraction are closed within the field



The curves only have integers as points

The points on the curve can form a cyclic group

The total number of points on the curve is called the order, and this is a prime number.

Multiplication of an integer k by a generator G leads to another point on the curve.

$$P = k * G$$

If k is zero, then it is said to resolve to a point at infinity.

Curves can have one or more cyclic subgroups.

$$n = h * r$$

n order of the curve

h Curve co-factor

r Order of the subgroups

Talk about subgroups

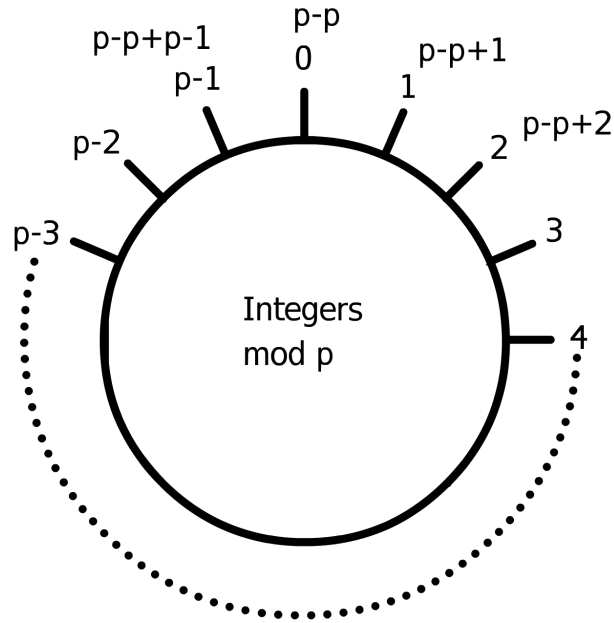
Groups in ECC

"An elliptic curve over a finite field can form a finite cyclic algebraic group [that is an order n that is prime]², which consists of all points on the curve."

[https://cryptobook.nakov.com/asymmetric-key-ciphers/
elliptic-curve-cryptography-ecc#order-and-cofactor-of-elliptic-curve](https://cryptobook.nakov.com/asymmetric-key-ciphers/elliptic-curve-cryptography-ecc#order-and-cofactor-of-elliptic-curve)

Groups in ECC

F_p p is a large prime number in a finite Field



Cryptographic Signatures: ECDSA

To generate a signature $\{ r, s \}$:

$$P = k * G$$

$$r = P_x \quad s = k^{-1}(\text{hash}(m) + d_v * P_x) \bmod p$$

k is a random secret number used once in the range $[0 \dots p-1]$

P_x is the x-coordinate of P

p is the order of the subgroup of the points generated by G

d_v is the private signing key

m is the message

G is the generator point

Signature is not deterministic due the random number k

Validate the signature:

$s_m = s^{-1} \bmod p$ is the modular inverse of s

$$R' = (\text{hash}(m) * s_m) * G + (r * s_m) * d_p$$

if $R'_x = P_x$ the signature is valid

d_p is the public key

Real World Cryptography, David Wong, Manning, pg. 143 - 144

<https://cryptobook.nakov.com/digital-signatures/ecdsa-sign-verify-messages>

<https://learn.saylor.org/mod/book/view.php?id=36341&chapterid=18920>

Cryptographic Signatures: EdDSA

To calculate the signature: $\{ R, s \}$

d_v private signing key generated from random integer (seed)

$d_p = d_v * G$ public key

$r = \text{hash}(\text{hash}(d_v + m)) \bmod q$

$R = r * G$

$h = \text{hash}(R + d_p + m) \bmod q$

$s = \text{hash}(r + h * d_v) \bmod q$

To validate the signature:

$h = \text{hash}(R + d_v + m) \bmod q$

$P_1 = s * G$

$P_2 = R + h * d_p$

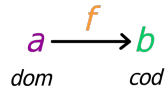
$P_1 = P_2 ?$

<https://cryptobook.nakov.com/digital-signatures/eddsa-and-ed25519>

Definition of a Category

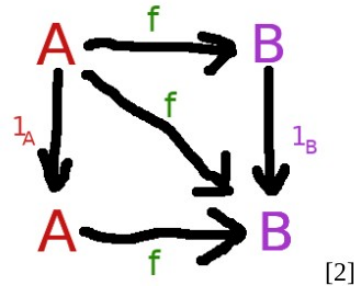
A category consists of:

- a collection of objects
- a collection of arrows



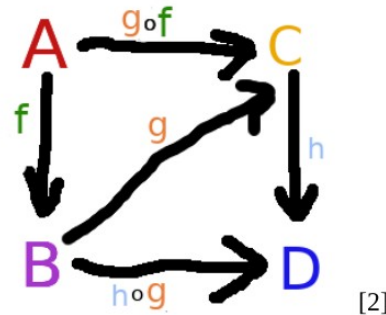
Identity:

$$f \circ 1_a = f = 1_b \circ f$$



Associativity:

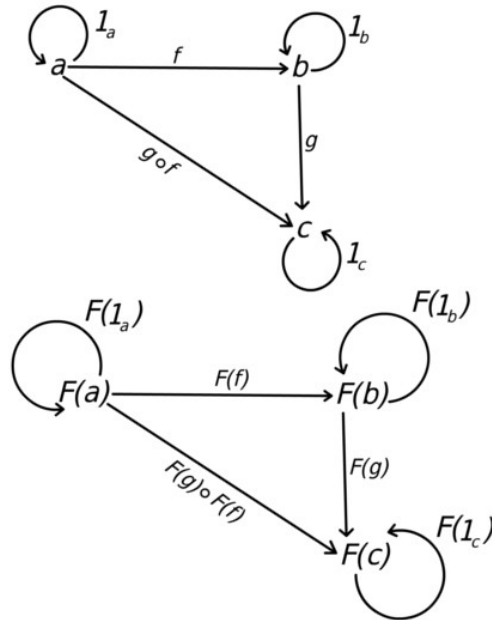
If morphism $A \rightarrow B$ is f , $B \rightarrow C$ is g , $C \rightarrow D$ is h then $A \rightarrow D$ is $(h \circ g) \circ f = h \circ (g \circ f) = h \circ g \circ f$



Uses of a Category

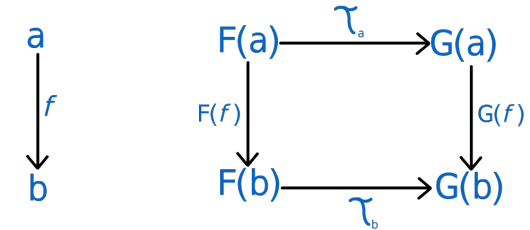
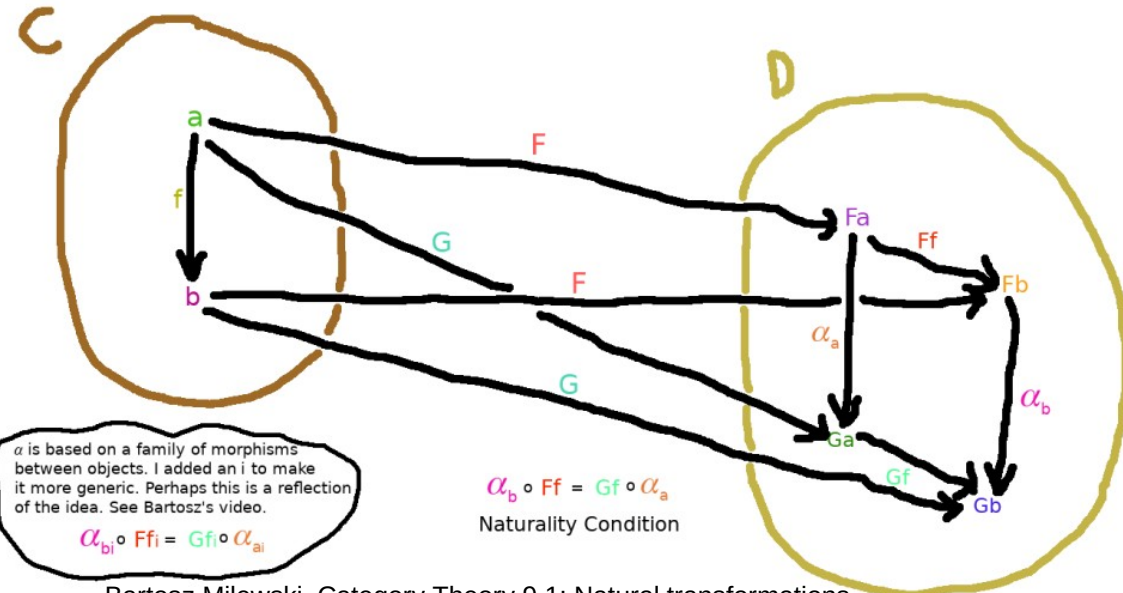
Definition of a Functor:

'A functor is a transformation from one category to another that "preserves" the categorical structure of its source'
pg. 194, The Categorical Analysis of Logic
-Goldblatt



Natural Transformations Consider Functors to be Objects
pg. 198, Goldblatt

Definition of a Natural Transformation:



pg. 199, Goldblatt

Groups as Categories

“In particular, a group is a category with one object, in which every arrow is an iso. If G and H are groups, regarded as categories, then we can consider arbitrary functors between them $f : G \rightarrow H$. It is obvious that a functor between groups is exactly the same thing as a group homomorphism.”
pg. 72, chap 4, Category Theory, Steve Adowey

Syntactic and Semantic Mappings

- Use RDF serializations like JSON-LD, JSON-Schema
SON Schema, Schema.org, JSON-LD: What's the Difference?
<https://dashjoin.medium.com/json-schema-schema-org-json-ld-whats-the-difference-e30d7315686a>
- Cryptographic proof of data:
<https://www.w3.org/TR/vc-data-integrity/>
→ Binary or RDF Canonicalization
<https://w3c-ccg.github.io/rdf-dataset-canonicalization/spec/index.html>

Burak Sedar's comments in e-mail about interop: In his opinion Identity does not consider semantics per se, but "LSA is a framework that aims semantic interoperability, so it offers some interesting ways to pass data through lenses. We do it by converting everything to labeled property graphs, and modifying graphs using openCypher in memory."

To him, working with RDF only ensures structural interoperability, but he went a level beyond and considered semantic interoperability. See the blog post: <https://cloudprivacylabs.com/blog/post-20220831/> and the paper: A Demonstration of Layered Schema Architecture as a Semantic Harmonization Tool, Burak Serdar, Cloud Privacy Labs <https://ceur-ws.org/Vol-3249/paper3-OSS.pdf>.

Solutions Out in the Wild

- What does FQL, CQL, and Hydra Do?

CQL supersedes FQL. Open source at: <https://www.categoricaldata.net/>, <https://github.com/CategoricalData>

"Open-source CQL and its integrated development environment (IDE) performs data-related tasks — such as querying, combining, migrating, and evolving databases — using category theory, a branch of mathematics that has revolutionized several areas of computer science. "

<https://github.com/CategoricalData/hydra> (Ryan Wisnesky and Josh Shnavier)

Hydra is based on Dragon. <https://eng.uber.com/dragon-schema-integration-at-uber-scale/> [Dragon: Schema Integration at Uber

Scale, Uber Engineering] -- → Provides Mappings between Data Models

and based on Algebraic Property Graphs in Shnavier et al., <https://arxiv.org/abs/1909.04881>

"Observing that algebraic data types are a common foundation of most of the enterprise schema languages we deal with in practice, for graph data or otherwise, we introduce a type theory for algebraic property graphs wherein the types denote both algebraic data types in the sense of functional programming and join-union E/R diagrams in the sense of database theory. We also provide theoretical foundations for graph transformation along schema mappings with by-construction guarantees of semantic consistency. "

Solutions Out in the Wild (Cont.)

- What does Project Cambria Do? <https://fission.codes/blog/project-cambria-overview/> (Focuses on how to read data in distributed systems, rather than using for writes to distributed systems. Converts: JSON, JSON PATCH, JSON Schema)
- What does Layered Schema Architecture Do? <https://layeredschemas.org/>, <https://github.com/cloudprivacylabs/lisa>
"LSA is a framework that aims semantic interoperability, so it offers some interesting ways to pass data through lenses. We do it by converting everything to labeled property graphs, and modifying graphs using openCypher in memory." – Burak Sedar
- What does Overlay Schema Architecture? <https://oca.colossi.network/guide/introduction.html>, Introduction to Overlays Capture Architecture (https://www.youtube.com/watch?v=D_Bqy4Mc514) . It is an architecture, not an ontology. It defines rules on how the data should be captured. Works as a data transformation pipeline.
→ Overlays Capture Architecture, a global solution for data capture and semantic harmonization. Uses a broadly defined Schema Base (like Hydra), and then builds cryptographically linked layers on top
- Benjamin Braatz Thesis (suggested by Ryan Wisnesky)
"While more sophisticated languages for the definition of ontologies have been developed, e. g., the Web Ontology Language (OWL), presented in [MH04] and related documents, we will only deal with RDF and its schema definition language RDF Schema in this thesis."

RDF/RDFS Reasoning:

<http://www.ksl.stanford.edu/software/JTP/doc/owl-reasoning.html>

