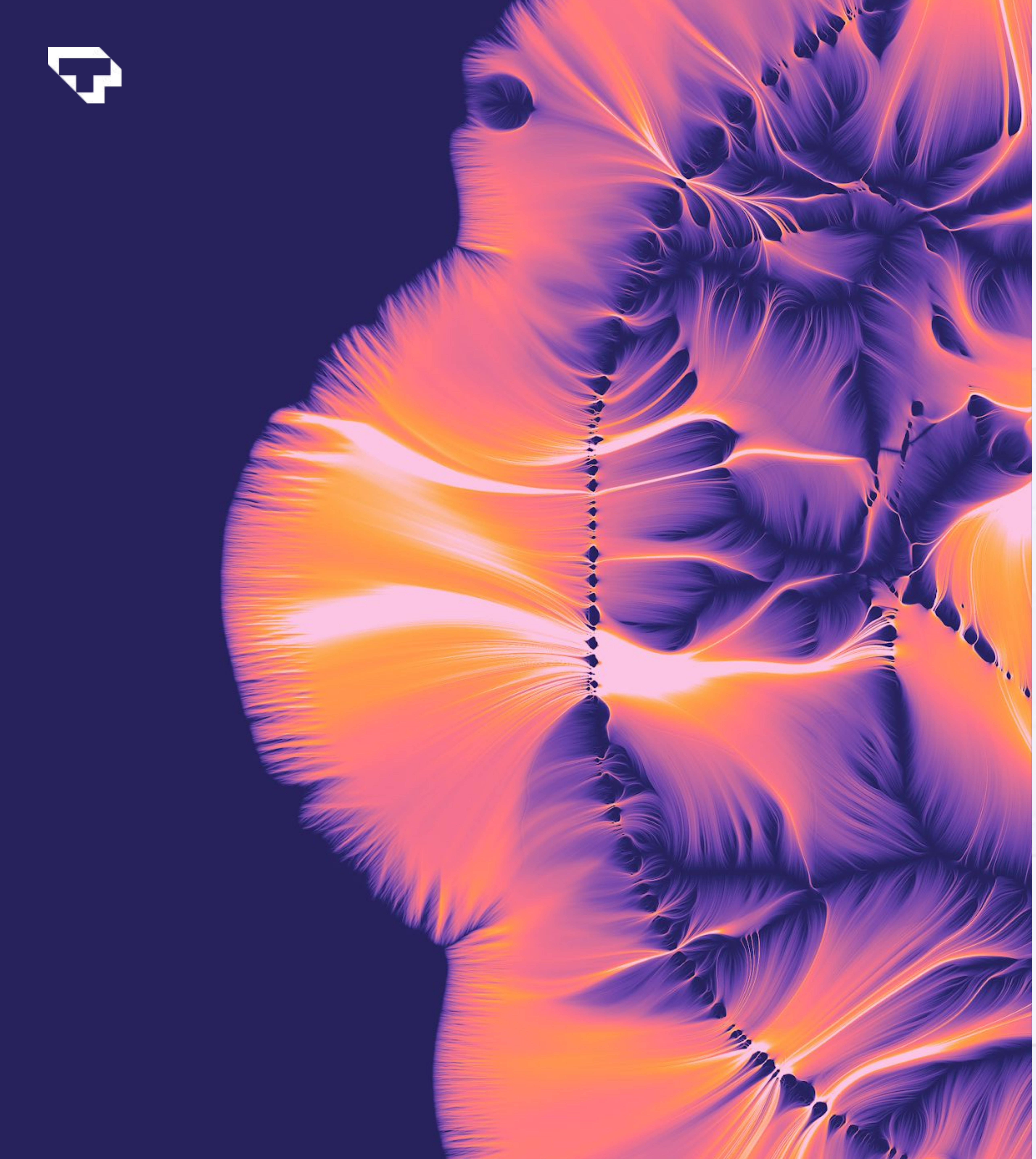

Nis Jespersen

Trust Graphs Design

Hands On Technical Session

About the Presenter

Nis Jespersen

- Solution Architect at Transmute
- Editor UN Linked Data Web Vocabulary
- nis@transmute.industries

Transmute

- Holds Editor and Author roles on the W3C Decentralized Identifiers and Verifiable Credentials Specifications
- Transmute contributes significant open source tooling in the Linked Data, DID and VC space

US DHS SVIP

- Transmute is contracted by US DHS SVIP to design Linked Data Verifiable Credential schemas, ensuring that appropriate UN terms are used in digital communication with the US government: https://w3id.org/traceability

Verifiable Data Platform

- Our Verifiable Data Platform guarantees standards compliance, sign up for free now and start building Trust Graphs immediately: https://platform.transmute.industries/

Ambition

- YOU are in the driver’s seat in this session, doing the work □
- YOU will design a data graph □
- YOU will issue a Verifiable Credential □

Agenda

LD

1. Select a suitable message from the business domain you are used to. This could be an instance from the previous API session or anything else, as long as touches some UN/CEFACT BSP data elements.

2. Build a small JSON example of such a domain message. To begin with, just make a quick'n'dirty example, using whatever property naming you're used to.

3. Go to https://vocabulary.uncefact.org/ and use the awesome search and filtering features find the UN/CEFACT definitions of some of the terms you were just using.

4. Add the UN/CEFACT LD context to your data and change the property names in your object accordingly.
   a. Confirm that you have build a data graph.
   b. Realize that you have changed your data message, broken your API contract.

5. Ditch the UN/CEFACT LD context and define your own context instead, so you can change back to your original API structure.
   a. Confirm that the resulting graph is the same
   b. Realize that your API contract is now intact
   c. Reflect upon the usage of your own vs a standard context

6. Extra credit: Include both contexts, make the graph change depending on the context order

7. Level 9000: Make your own external contect, put it somewhere on the internet and reference it from your LD file.

VC

1. Take your well-formed Linked Data and make it the credentialSubject of a Verifiable Credential.

2. Verify the Verifiable Credential.
   a. Reflect upon what it means that the VC verifies. What does it say about the relationship between the issuer and the credentialSubject?

3. Fiddle with the data, confirm that the Verifiable Credential is no longer valid.

4. Add some more LD elements to the credentialSubject and re-issue. The main ones are:
   a. @id field to the credentialSubject.
   b. @type field to the credentialSubject.
   c. What does this new information say about the verifiable data?

Tools of the Trade

UNECE

LD

Our vocabulary!

- https://vocabulary.uncefact.org/

Linked Data:

- https://v.jsld.org/
- https://json-ld.org/playground/

Graph Viewer:

- https://issemantic.net/rdf-visualizer

Graph Database:

- https://neo4j.com/cloud/platform/aura-graph-database/

VC

Issue and Verify

- https://platform.transmute.industries/
- https://api.did.actor/