

A wireframe model of an airplane, rendered in a glowing cyan color, is shown in profile against a dark blue background. The model highlights the fuselage, wings, and tail section with a network of lines and points.

Demo 2: Specimen Test Suite (STS)

FINAL DISSEMINATION EVENT

NLR Amsterdam (NL), December 17th

NLR: **Julian De Marchi** | Thomas Timmerman | Florian Hensel | Nikita Noskov | Carlos Schyns |
Annemoon De Groen | Wesley Blaauw

TUD: Francisco Monticeli | Davide Biagini | John-Alan Pascoe UT: Matias Lasen



Funded by
the European Union

Index

§ 01p03

Workflow

§ 02p10

Data backbone

§ 03p16

STS

§ 04p18

Future steps

01

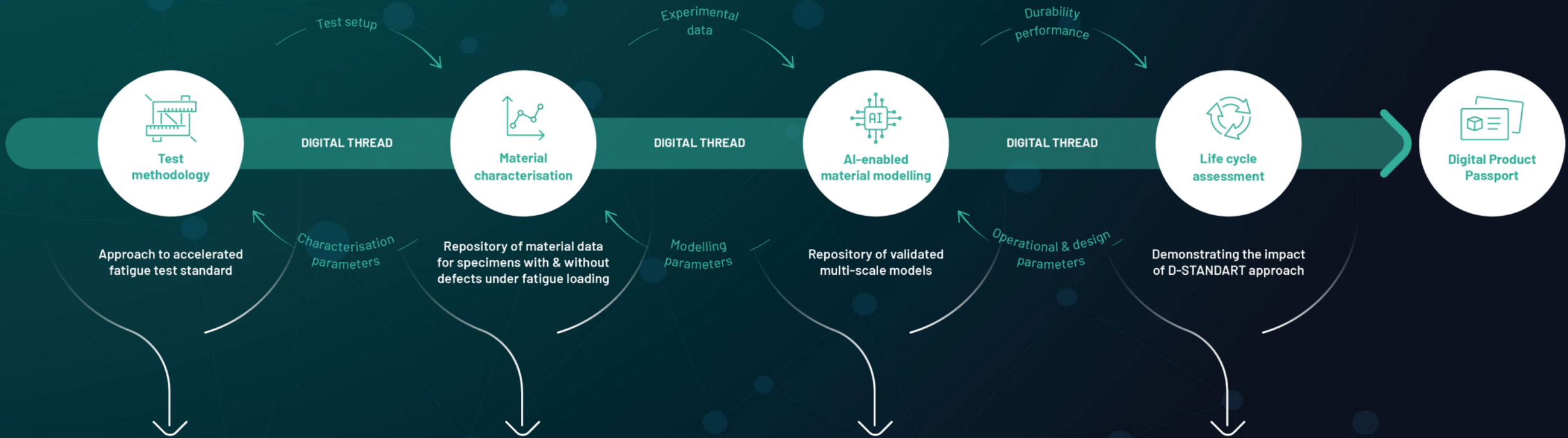
workflow

Materials characterization



D-STANDART approach

Workflow



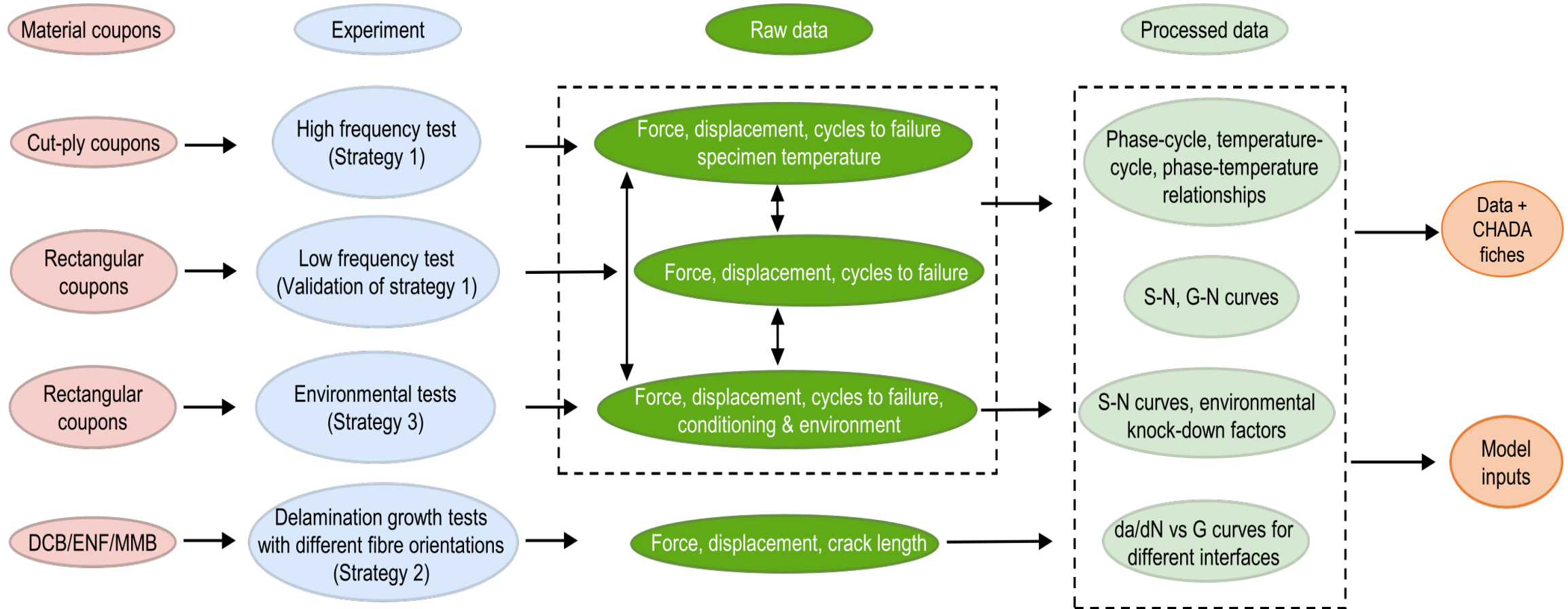
Digital Thread as data collection backbone

- › **Capturing characterization data**
 - Original CHADA for D-STANDART
 - Original MODA for D-STANDART
 - CHAMEO online
 - EMMO

Section 1

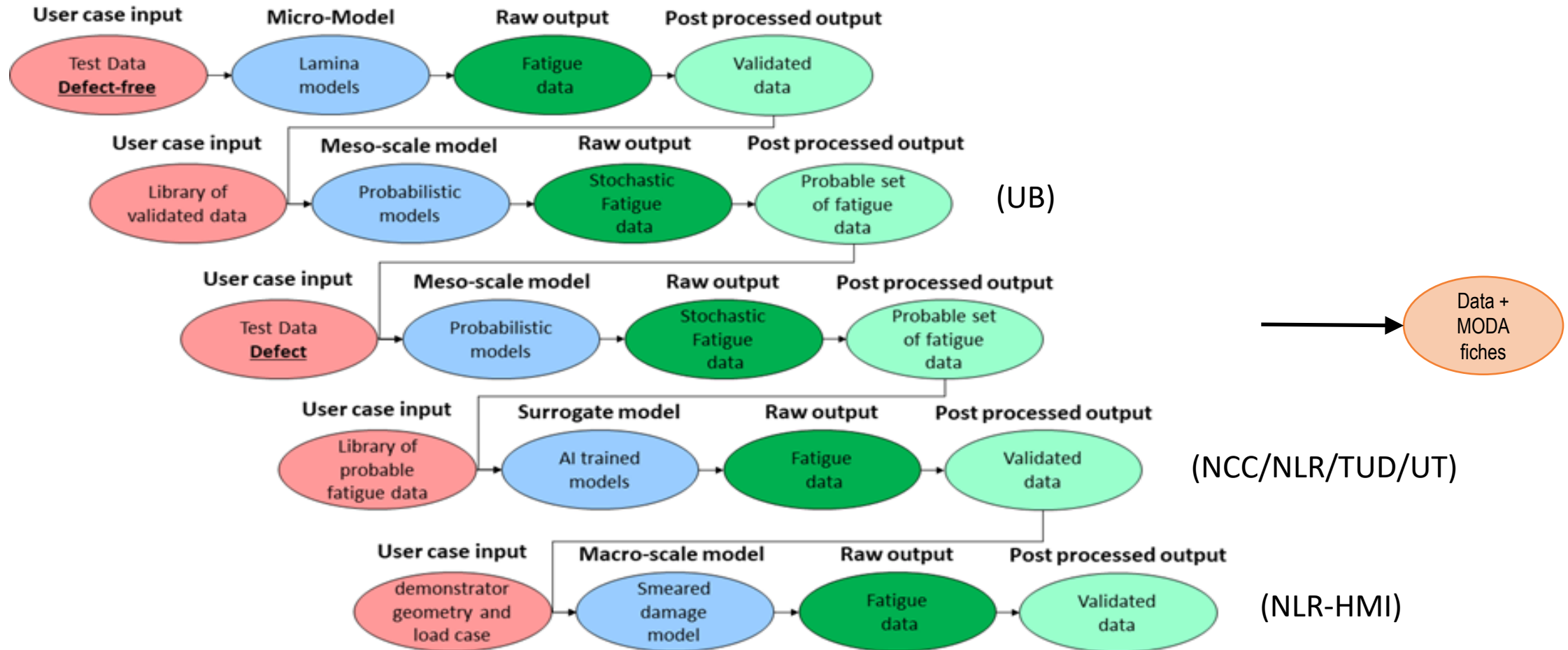
digital thread: CHADA fiche data capture

Workflow



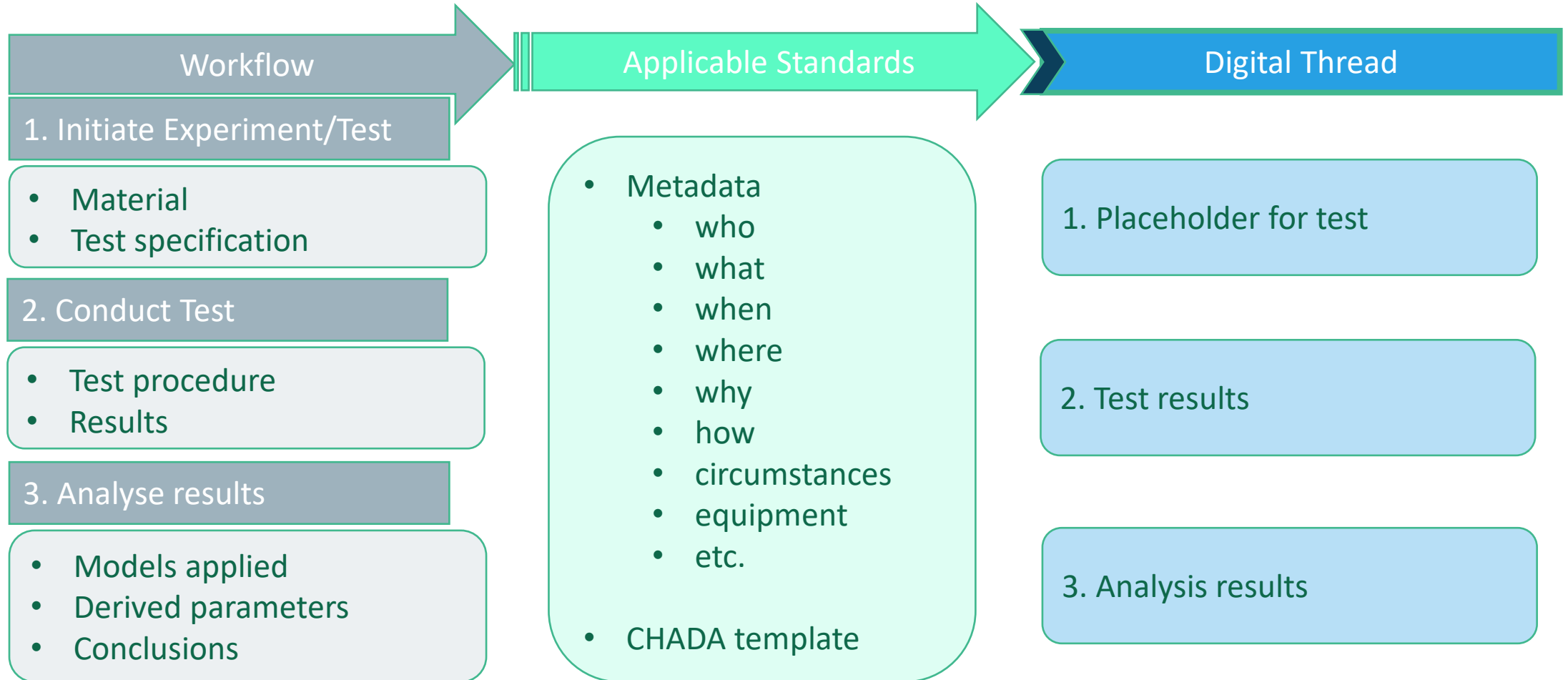
digital thread: MODA fiche data capture

Workflow



standards-based data consolidation

Workflow



02

Data backbone

Demo



CHADA

Data Backbone

› EMMC: European Materials Modelling Council

- emmc.eu

› **CHADA**: materials characterization data

- EU Materials Characterisation Council (EMCC: CWA17815:2025)

› **MODA**: materials modelling data

- EU Materials Modelling Council (EMMC: CWA17284:2018)

› EMMO: Elementary Multi-perspective Material Ontology

- github.com/emmo-repo/EMMO

› **CHAMEO**: characterization methodology domain ontology

- github.com/emmo-repo/domain-characterisation-methodology

› **CHOCO**: characterization of composites ontology

- developed by NLR, but not yet published

› Linking to ontology

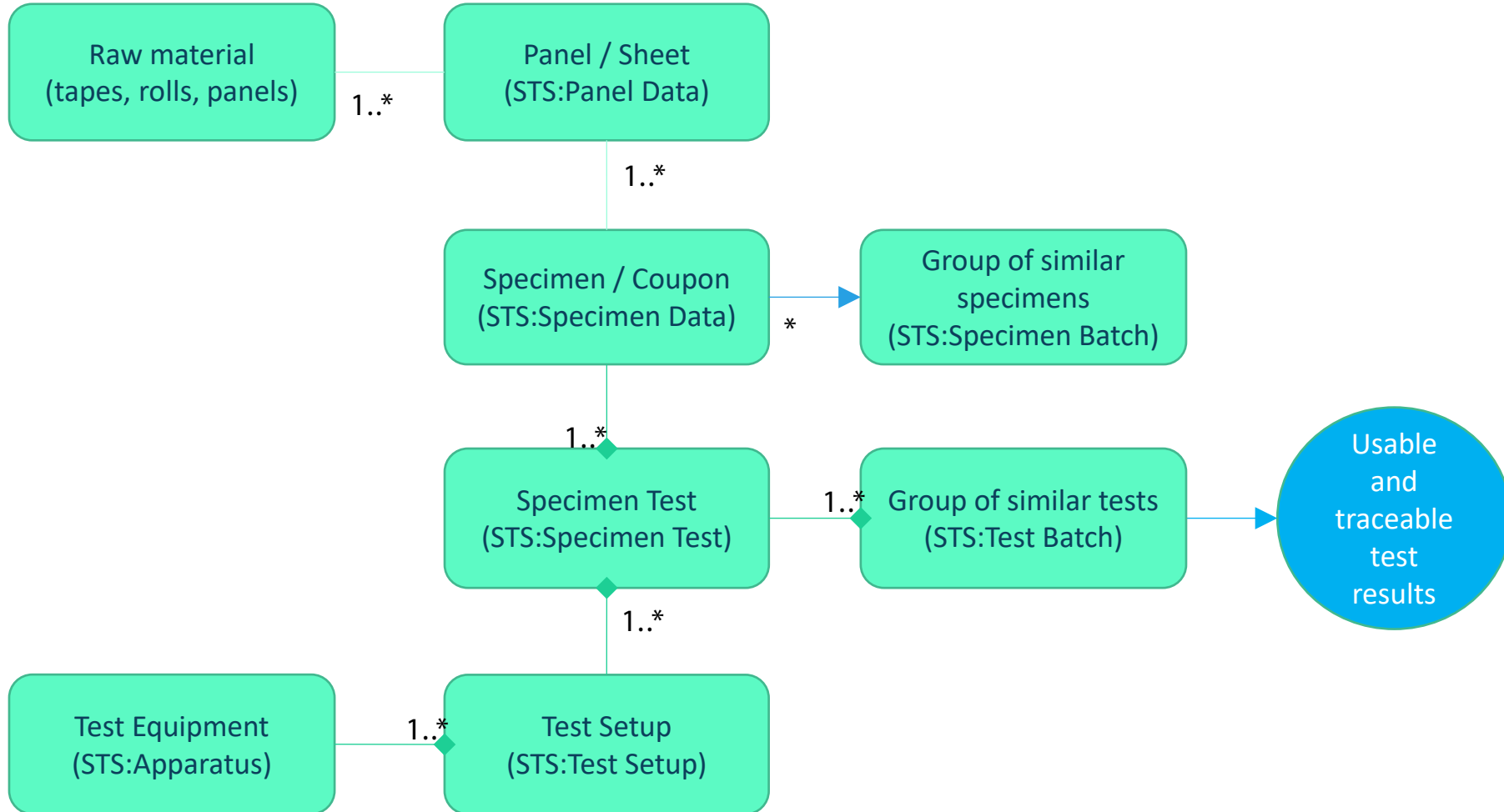
- NLR Master Sheet
- NLR CHOCO
- Knowledge graphs and RDF
- Currently still using RDBMS

› Importing data

- premise: CHADA as input
- premise: native data formats
- challenge: Sheer number of Zenodo (meta)data sets, Statistics
- shift: CHADA as output, STS as browser
- approach: IRI tool to capture metadata for input
- challenge: Variety of Zenodo (meta)data sets
- IRI for CHADA
- IRI for Excel metadata
- JSON-LD data format
- challenge: Variety & inconsistency in CHADAs
- shift: Back to manual input via STS
- invitation to provide expert input to matCHMaker consortium in 2026

data architecture

Data Backbone



03

STS

Demo



Section 1

specimen test suite web portal (sts.nlr.nl)

Demo

- › **dashboard**
- › **apparatus / equipment**
- › **test setup**
- › **panel / plate**
- › **specimen / coupon batch**
- › **specimen / coupon test**
- › **test batch**
- › *(aggregated test results)*

04

Triple store

Demo



Knowledge Graph – SPARQL Query

```
PREFIX choco: <https://example.org/choco#>
```

```
SELECT ?apparatus ?setup
WHERE {
  ?apparatus a choco:apparatus .
  ?apparatus choco:used_in_setup ?setup .
}
```

Relational DB – SQL Query

```
SELECT a.id AS apparatus, s.id AS setup
FROM apparatus a
JOIN apparatus_setup as_rel ON a.id = as_rel.apparatus_id
JOIN setup s ON as_rel.setup_id = s.id;}
```

As you can see, to find data from entities of 2 different is much harder in SQL, where we need to know the specific structure of the database

- › Remember that in an **Ontology** classes, properties, and rules are defined
- › **@Context:** Header that provides namespaces for the used vocabularies **1**
 - › **Choco:** Our Ontology
 - › **RDF:** The core data model for describing data as triples
 - › **RDFS:** Defines classes, properties, hierarchies
 - › **XSD:** Provides datatypes for literal values
- › **@Graph:** Data to be modeled as graph - class **2** & property definitions **3**
 - › **@id:** The unique identifier of each entity
 - › **@type:** Declares whether the entity is a class or property

Additionally for Property Definitions

- › **Rdfs:domain:** Specifies the class of entities the property applies to
- › **Rdfs:range:** The type of possible values this entity can have

```

"@context": {
  "choco": "https://example.org/choco#",
  "rdf": "http://www.w3.org/1999/02/22-rdf-syntax-ns#",
  "rdfs": "http://www.w3.org/2000/01/rdf-schema#",
  "xsd": "http://www.w3.org/2001/XMLSchema#"
},
{
  "@id": "choco:apparatus",
  "@type": "rdfs:Class",
  "rdfs:label": "Apparatus",
  "rdfs:comment": "Class Description"
},
{
  "@id": "choco:setup",
  "@type": "rdfs:Class",
  "rdfs:label": "Setup",
  "rdfs:comment": "Class Description"
},
{
  "@id": "choco:panel",
  "@type": "rdfs:Class",
  "rdfs:label": "Panel",
  "rdfs:comment": "Class Description"
},
{
  "@id": "choco:docs",
  "@type": "rdf:Property",
  "rdfs:label": "Additional Documents",
  "rdfs:domain": "rdfs:Resource",
  "rdfs:range": "xsd:anyURI"
},
{
  "@id": "choco:description",
  "@type": "rdf:Property",
  "rdfs:label": "Description",
  "rdfs:domain": [ "choco:apparatus", "choco:setup" ],
  "rdfs:range": "xsd:string"
},
{
  "@id": "choco:used_in_setup",
  "@type": "rdf:Property",
  "rdfs:label": "Used in Setup(s)",
  "rdfs:domain": "choco:apparatus",
  "rdfs:range": "choco:setup"
}

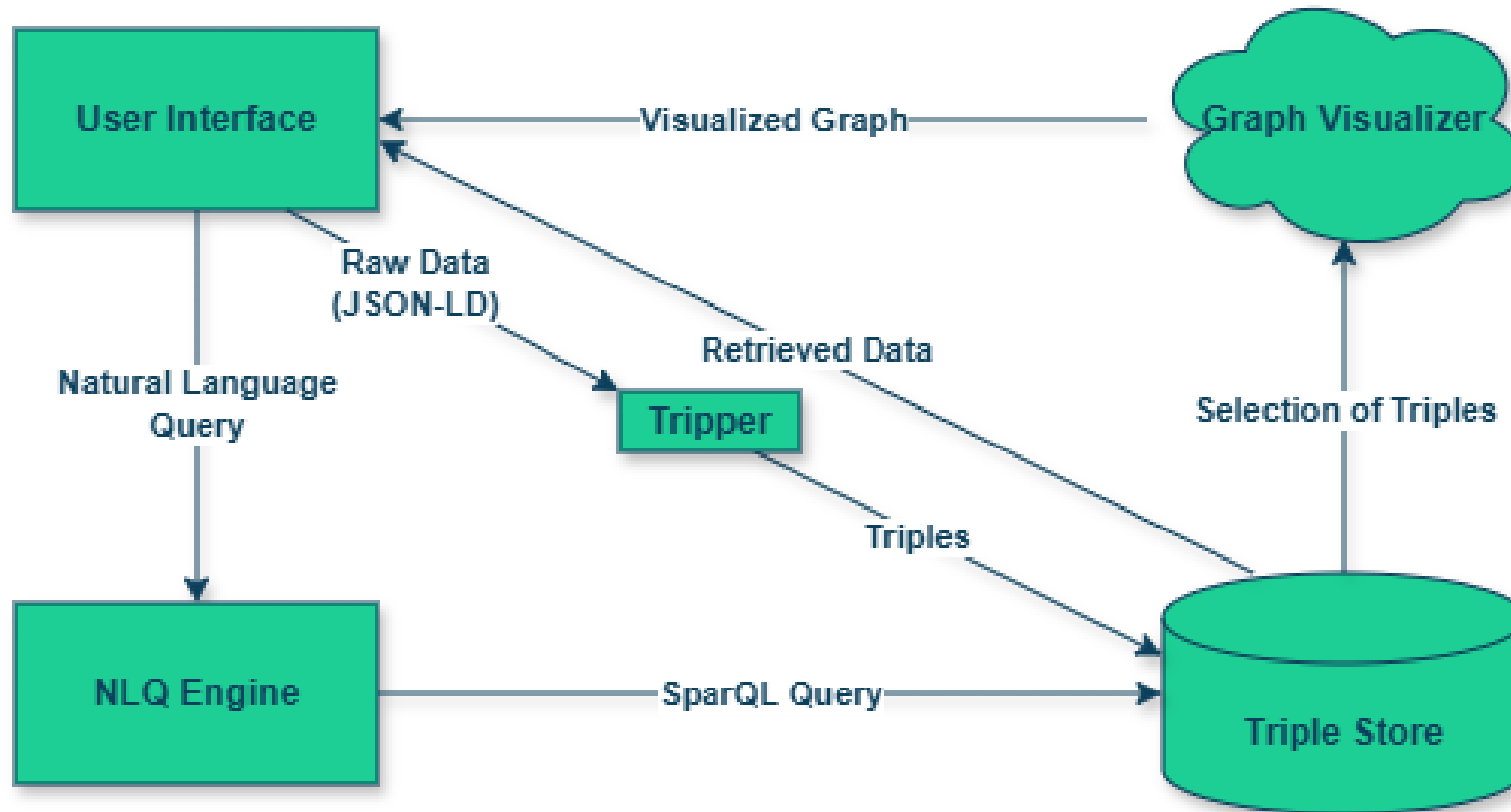
```

- › **Instance data** represents real-world entities with attributes and relationships (choco:used_in_setup, choco:used_to_test_specimen)
- › **@id** gives each instance a **unique IRI**
 - › if omitted, the system generates a blank node ID
- › **Not all fields for an entity have to be filled in!**
 - › Entering empty fields will result in triples with empty values
- › **Avoid using empty string literals unless intentional;** better to omit the property
- › Triple stores usually accept all triples, even if the datatype clashes
 - › Reasoning/validation may flag errors

```
{
  "@context": {
    "choco": "https://example.org/choco#"
  },
  "@graph": [
    {
      "@id": "choco:apparatus/NCC::SONY%20ProShot",
      "@type": "choco:apparatus",
      "choco:uid": "NCC::SONY ProShot",
      "choco:local_id": "SONY ProShot",
      "choco:org_id": "NCC",
      "choco:org_user": "Kieron Bancroft",
      "choco:publication_date": "2024-06-06",
      "choco:publication_access": "public",
      "choco:used_in_setup": "NCC::Instrom 5985 (250 kN)",
      "choco:signal_resolution": 5.0,
      "choco:notes": "Digital image correlation used for strain measurement"
    },
    {
      "@id": "choco:setup/NCC::Instrom%205985%20(250%20kN)",
      "@type": "choco:setup",
      "choco:uid": "NCC::Instrom 5985 (250 kN)",
      "choco:local_id": "Instrom 5985 (250 kN)",
      "choco:org_id": "NCC",
      "choco:org_user": "Kieron Bancroft",
      "choco:publication_date": "2024-06-06",
      "choco:publication_access": "public",
      "choco:used_to_test_specimen": "NCC::P5-QI-T-01",
      "choco:description": "clamp #2716-015"
    },
    {
      "@id": "choco:test_data/NCC::P5-QI-T-01",
      "@type": "choco:test_data",
      "choco:uid": "NCC::P5-QI-T-01",
      "choco:local_id": "P5-QI-T-01",
      "choco:org_id": "NCC",
      "choco:org_user": "Kieron Bancroft",
      "choco:publication_date": "2024-07-06",
      "choco:publication_access": "public"
    }
  ]
}
```

potential Infrastructure

Triple Store



- › Apache Jena is **a framework** for working with RDF data, ontologies, and SPARQL
 - › Apache Jena TDB is the **Triple Store**, a graph database that stores triples
 - › Apache Jena Fuseki is the **SPARQL Server** which allows external querying of data
- › **Best for lightweight RDF applications**, flexible ontology management, and academic research
- › **Fully open-source solution**
- › **Tripper integration:** Python package developed by EMMO. Users upload data to frontend, Tripper converts it into RDF triples and uploads it to the Triple Store



- › **Graph visualizers:**
 - › **Cytoscape.js** or **Gephi** — open-source, run locally, can render subsets of RDF instance data from Jena
<https://js.cytoscape.org/>
<https://gephi.org/>
- › **Open-source NLQ for SPARQL:** lets users ask questions in natural language, automatically translated into SPARQL queries
 - › **Sparklis** — interactive natural language querying over SPARQL endpoints
<https://github.com/sebferre/sparklis>
 - › **QAnswer** (community edition) — can be deployed locally, translates natural language into SPARQL
<https://doc.qanswer.ai/>

Graph visualizers can show the ontology / instance data, while NLQ tools help them query it without needing SPARQL expertise. More research here is needed!

- › **For D-STANDART** we can leverage the inference capabilities of KGs as well as NLQ and graph visualization
- › **Apache Jena + TDB** as triple store
 - › Use tripper (EMMC) to input JSON-LD, transform it to RDF triples, and automatically store them in the database
- › **Sparklis** or **Qanswer** as potential NLQ engines
 - › More research is needed
- › **Cytoscape.js** or **Gephi** as potential graph visualizers
 - › More research is needed

Questions for the audience

Regarding the anticipated D-STANDART digital thread data and fatigue prediction capabilities



- › Do you (think you will) use digital threads and/or digital product passports within your own organization?
- › If so, what do you use them for and how do you apply them to your product design and fabrication processes?
- › How do you think you might be able to apply the D-STANDART digital thread and fatigue prediction models at your organization?
- › What data might you actually need and use from the D-STANDART digital thread, and in what format(s)?
- › What insights would expect the D-STANDART fatigue prediction models to provide? And how would you want that presented?
- › How can we ensure that the D-STANDART digital thread and fatigue prediction models interconnect with your digital ecosystem?
- › D-STANDART will develop a plug-in for a CAE tool for fatigue modelling; would you use this, and if so, how should it work?
- › How can we help ensure that the D-STANDART digital thread can be seamlessly integrated with your own digital ecosystem?

Thank you!

Contact points for any question:

FOLLOW US ON:



zenodo

d-standart.eu



› Presenter

- › Julian DE MARCHI (NLR)
- › Email: Julian.DeMarchi@nlr.nl
- › Phone: +31 88 511 3792

› Project Lead

- › Marco NAWIJN (NLR)
- › Email: Marco.Nawijn@nlr.nl
- › Phone: +31 88 511 4486

› Project Management Officer

- › Mathieu LIONS (LUP)
- › Email: Mathieu.Lions@l-up.com
- › Phone: +33 7 83 28 19 82

› Communication Officer

- › Peggy FAVIER (LUP)
- › Email: Peggy.Favier@l-up.com
- › Phone: +33 6 75 64 10 78



Funded by
the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.



UK Research
and Innovation

