ONE Record Insights

Episode 3

Crafting Ontologies: From physical freight to machine readable data

René Pietzsch
Head of Product Mgt, Linked & Semantic Data Consultant & Scrum Master, eccenca

Andra Blaj
Developer, ONE Record, IATA
Your hosts today

René Pietzsch
Head of Product Management, Linked & Semantic Data Consultant & Scrum Master, eccenca

Andra Blaj
Developer, ONE Record IATA
The data model: a digital twin of the air cargo industry
Tuesday, 30th June 11:00 – 12:30 (CEST)

Crafting ontologies: from physical freight to machine readable data
Tuesday, 7th July 11:00 – 12:30 (CEST)

The ONE Record API: an overview of the key features
Tuesday, 14th July 11:00 – 12:30 (CEST)

Data security: securing the Internet of Logistics
Tuesday, 21st July 11:00 – 12:30 (CEST)

Pilot testing: engaging with the cargo community
Tuesday, 28th July 11:00 – 12:30 (CEST)
ONE Record Insights

Episode 2
The data model: a digital twin of the air cargo industry
Tuesday, 30th June 11:00 – 12:30 (CEST)

Episode 3
Crafting ontologies: from physical freight to machine readable data
Tuesday, 7th July 11:00 – 12:30 (CEST)

Episode 4
The ONE Record API: an overview of the key features
Tuesday, 14th July 11:00 – 12:30 (CEST)

Episode 5
Data security: securing the Internet of Logistics
Tuesday, 21st July 11:00 – 12:30 (CEST)

Episode 6
Pilot testing: engaging with the cargo community
Tuesday, 28th July 11:00 – 12:30 (CEST)
Crafting Ontologies: From physical freight to machine readable data

Part 1  History and context

Part 2  Let's craft an ontology

Part 3  Operationalizing ontologies

Part 4  The ONE Record ontology

Part 5  Q&A
How to participate during the meeting?

• You can only hear the presenters

• Your microphones are disabled

• Use the questions box to interact

• Simply enter your questions in the chat box on the right
This meeting is recorded for future use

The entire recording along with questions will be available shortly after this webinar finishes.

Simply click on the link in the invite for the live event to access it.

The presentations shown today will be available for download on our website:

www.iata.org/one-record
ONE Record Insights

Part 1

History and context
The Invention of the Internet
(Semantic Web) Standards

User Interface & applications

- Trust
- Proof
- Unifying Logic

Query: SPARQL
Ontology: OWL
Rules: RIF
RDF-S

Data-interchange: RDF

XML

URI
Unicode

Semantic Web Layer Cake, 2006
(Semantic Web) Standards
RDF, RDFS, OWL... WTF...

Diagram showing the relationships between Jimmy Page, Person, Led Zeppelin, Album, Song, Creative Work, and concepts such as "Led Zeppelin IV" and "Black Dog."
RDF - Resource Description Framework
RDF - Resource Description Framework
RDF - Resource Description Framework

Diagram showing relationships between entities such as Jimmy Page, Person, Led Zeppelin, Band, Album, Song, and Creative Work. The diagram includes relationships like "is a", "has member", "records", "has track", and "release date".

Diagram taken from: https://zenodo.org/record/3898519?hsCtaTracking=870ee1df-0d7b-4e50-8767-0b79d2a8295f0%7C40a224e4-9b70-4886-9520-95036d56f67e#.XuoMVy97HRY
RDF - Resource Description Framework

![RDF Diagram]

Diagram taken from https://www.deeplearn.org/3848157/ThisIsTracking-d70eef-847-4e0-872-78/deeplearn.org/744a82d4-462-486-852-9855056d7e#.XuoMVy5v97HYR
RDF - Resource Description Framework

![RDF Diagram](https://zenodo.org/record/3898519?hsCtaTracking=870ee1df-0d7b-4e50-8767-b79d2a8295f0%7C40a224e4-9b70-4886-9520-95036d56f67e#.XuoMVy97HRY)
RDF - Resource Description Framework
RDF - Resource Description Framework
RDFS - RDF Schema

_taken from https://www.dbpedia.org/ontology/Place

Led Zeppelin records "Led Zeppelin IV" on an album that has the release date of 8/11/1971. This album contains the song "Black Dog" which is a creative work.

Jimmy Page is a member of the band Led Zeppelin, which is a type of creative work. The song "Black Dog" is a type of creative work.
RDFS - RDF Schema

Diagram showing relationships between entities:
- Jimmy Page is a Person
- Led Zeppelin is a Band
- Led Zeppelin IV is an Album
- "Black Dog" is a Song
- "Black Dog" has track on Led Zeppelin IV
- 8/11/1971 is the release date of Led Zeppelin IV

Diagram taken from: https://zenodo.org/record/3898519?hsCtaTracking=870ee1df-0d7b-4e50-8767-b79d2a8295f0%7C40a224e4-9b70-4886-9520-95036d56f67e#.XuoMVy97HRY
OWL - Web Ontology Language

Diagram showing relationships between entities such as Jimmy Page, Person, Led Zeppelin, Band, Album, Song, and Creative Work.
Linked (Open) Data Principles
The Growth of the LOD Graph
Linked Open Data, 2007

Image shows datasets that have been published in Linked Data format, by contributors to the Linking Open Data community project and other individuals and organizations.
The Growth of the LOD Graph
Linked Open Data, 2009
The Growth of the LOD Graph
Linked Open Data, 2014
The Growth of the LOD Graph

Linked Open Data, 2020
Part 2

Let’s craft an ontology
Introduction

Ontologies are shared, structured conceptualizations of domains.

Conceptualization:
• abstract, simplified view of the domain represented for some purpose
• captures objects, concepts and the relationships that hold among them

Structured:
• According to standardized modelling paradigms like OWL and SKOS

Shared:
• Agreed upon by domain experts and acting as an extensible standard for data representation in the domain

Domains vary from most basic things (Persons, Places, Activities) to specialist areas like supply chains or manufacturing
Wait ... “Google” for prior Work and / or Inspiration ... LOV!
Engineering Methodology

• No one correct way to ontology engineering, depends on the domain and application

• Generally:
  • Concepts in the ontology should be close to objects (physical or logical) and relationships in your domain of interest.
  • Most likely, they are nouns (objects) and relations (verbs) in sentences that describe your domain.
  • Additionally, the scope of use and application determine which objects and relations we need to define

Necessarily iterative – created ontologies have to be discussed, applied and refined
Scope and requirements

• Scope depends on domain, application and use of the ontology
• Is there already data available, or are we modelling “on the green field”?
• If there is data available, what is it used for?
• Are we trying to create an easy to understand, logical data model that fits the domain?
• Are we trying to adhere to existing structures for legacy compatibility?
• How much meta data is needed?
• What other data sources are relevant?
• Are there already ontologies out there, that model the domain?

Useful technique: **Ontology competency questions**
• Create a catalog of questions your ontology should answer
• After each development milestone / iteration, try to answer the questions by:
  • Creating example instance data according to your ontology that you can query or browse with the goal to answer the question
Source material

- Source material can be wide-ranging:
  - Existing, relational data sources (Excel, databases etc)
  - Slide decks that present the domain / train new users on existing concepts
  - Company taxonomy databases
  - Product catalogs
  - Books on the topic in question
  - Expert interviews

- Often a combination of these:
  - Receive a spreadsheet and a slide deck
  - Interview a domain expert (or be one yourself)
  - Create an initial ontology, then iterate with the expert
  - Receive more documentation material etc.
Example Domain: Product Data

Our company’s physical products and services are managed by product managers. The practice of product management is distributed across all department in which they have a direct report. Products are structured into Categories for which expertise exists in product managers and responsible departments.

E.g. our P516-8211068 named Film Multiplexer Rheostat Warp is managed by Kristen Bauers in Procurement.
Our company physical products and services are managed by product managers. The practice of product management is distributed across all department in which they have a direct report. Products are structured into Categories for which expertise exists in product managers and responsible departments.

E.g. our P516-8211068 named Film Multiplexer Rheostat Warp is managed by Kristen Bauers in Procurement.

1. Find concepts
Our company physical products and services are managed by product managers. The practice of product management is distributed across all department in which they have a direct report. Products are structured into Categories for which expertise exists in product managers and responsible departments.

E.g. our P516-8211068 named Film Multiplexer Rheostat Warp is managed by Kristen Bauers in Procurement.

1. Find concepts
2. Find relationships
Example Domain: Product Data

Our company physical products and services are managed by product managers. The practice of product management is distributed across all department in which they have a direct report. Products are structured into Categories for which expertise exists in product managers and responsible departments.

E.g. our P516-8211068 named Film Multiplexer Rheostat Warp is managed by Kristen Bauers in Procurement.

1. Find concepts
2. Find relationships
3. Find instances (typically in data sources, not in documentation material)
Our company physical **products** and **services** are **managed by** **product managers**. The practice of product management is distributed across all **department** in which they **have a direct report**. **Products** are **structured into Categories** for which **expertise exists in product managers and responsible departments**.

E.g. our **P516-8211068** named **Film Multiplexer Rheostat Warp** is managed by **Kristen Bauers in Procurement**.

1. Find **concepts**
2. Find **relationships**
3. Find **instances** (typically in data sources, not in documentation material)
4. Find **attributes**

Manufacturing: Define classes

• Methodologies:
  • top-down: Define most general classes first, then specialize
  • bottom-up: Define most specific classes, then group them to generalize
  • middle-out: Define most useful, salient classes first, then specialize and generalize as appropriate

• Class candidates: product, service, product manager, department, direct report, categories

• Starting with these classes, often helpful to draw a diagram, presenting classes as boxes and relations as lines
Manufacturing: Define classes

- Class candidates: product, service, product manager, department, direct report, categories
Manufacturing: Define classes

- Class candidates: product, service, product manager, department, direct report, categories
Manufacturing: Define classes

- Relation candidates: managed by, has (direct report), structured (into category), expert in, responsible for
Manufacturing: Define classes

- Relation candidates: managed by, has (direct report), structured (into category), expert in, responsible for
Manufacturing: Define classes

- Attribute candidates: named

- category
  - has category
  - manages product
- product
  - name: string
- hardware
- service
- agent
  - expert for
- employee
  - manages product
- manager
- department
  - direct report
Manufacturing: Define classes

- Attribute candidates: named

[Diagram showing relationships between classes and attributes such as product name, id, has category, expert for, manages, department name, etc.]
Manufacturing: Formalize Protégé Ontology editor
Manufacturing: Formalize using (Excel) Template and automatic mapping into RDFS/OWL
Links

- **Book:** Dean Allemang, Jim Hendler - *Working Ontologist*
- **Web:** Leigh Dodds, Ian Davis - *Linked Data Pattern*
Git / CI CD of ontology

- Treat ontologies like code -> [git icon]
- Apply automatic integration and test procedures
- Automatic generation of visualization and documentation artifacts
Testing

• Instance data can be tested using RDFUnit as presented in a previous lecture
• This testing can be automated for continuous integration
2. Products vocabulary: Overview

This ontology has the following classes and properties.

Classes
- Agent
- Department
- Employee
- Hardware
- Manager
- Price
- Product
- ProductCategory
- Service

Object Properties
- eligible for
- expert for
- has category
- has direct report
- has manager
- hasproductmanager
- manages product
- member of
- price
- responsible for

Data Properties
- address
text
- amount
- area of expertise
currency
- depth in mm
- email
- height in mm
- ID
- name
- phone number
- weight in g
- width in mm

Annotation Properties
- created
- creator
- issue
- license
- modified
- name
- preferred namespace
- prefix
- preferred namespace
- uri
- publisher
- rights
- vocabulary

Named Individuals
- ecenca gmb h
- pietzsch

3. Products vocabulary: Description

This is a placeholder text for the description of your ontology. The description should include an explanation and a diagram explaining how the classes are related, examples of usage, etc.

4. Cross reference for Products vocabulary classes, properties and dataproperties
ONE Record Insights

Part 3

Operationalizing ontologies
IATA: One Record Data Model
APICS + eccenca: SCOR-VOC

The value in the rectangle represents a SCOR Metric result. For example: Orders delivered in time for certain goods.
The universal Semantic Data Collaboration Paradigm
But ... How ...?
But ... How ... Now!

• Low level RDF libraries
• Mapping frameworks and mapping languages
• Triple Stores
• Enterprise knowledge graph platform
eccenca Corporate Memory an Enterprise Knowledge Graph Platform
Corporate Memory: User Journey and Functional Areas

Build
- Data Ingestion
- Workspace Interface
- Dataset Management
- Workflow Editor
- File & DB Connectors
- Mapping Editor
- Linking Editor

Knowledge Graph

Explore
- Vocabulary Catalog
- Taxonomy Management
- Graph Exploration
- Data Shape Editor
- Query Catalog
- OWL Visualization

Data Shopping

Consume
- Target File Types
- SPARQL Endpoint
- SQL Endpoint
- Graph Store API
- RDF Resource API (CBD)
- JSON-LD Frame API

Govern
- Data Workflows
- APIs
- ACL Graph
- Query Rewriting
- Logging / Monitoring
ONE Record Insights

Part 4

The ONE Record ontology
ONE Record Data Model Components
Determine the domain and scope of the ontology

What is the domain that the ontology will cover?

What information the ontology should cover?

How can we share this ontology?

For what we are going to use the ontology?

Who will create, use and maintain the ontology?
To support the deployment and the adoption of the ONE Record Data Model, IATA published a set of specification, guidance materials and tools.

Data Model: Standard components

To support the deployment and the adoption of the ONE Record Data Model, IATA published a set of specification, guidance materials and tools

Ontology Creation: Tools & Methodology
Ontology Creation with Protégé

Developed by the Stanford Center for Biomedical Informatics Research at the Stanford University School of Medicine, Protégé tool is one of the oldest and most widely deployed ontology modelling tools. It was originally conceived as a frame-based modelling tool for rich ontologies following the Open Knowledge Base Connectivity protocol. Later iterations of Protégé have expanded to include a plug-in that is now widely used for OWL and RDF modelling.

https://protege.stanford.edu/
Creating an ontology

Iterative process

- Models & Terms
- Classes
- Properties
- Turtle representation
Models & Terms

- Transport Means
- Transport Segment
- ULD
- Item
- Piece
- Shipment
- Waybill
- Booking
- Quote Request
- Offers

- Product
- Can be in → Piece
- Can be of ↓
- Is of ↓
- Is of
- Allocated on →
- Is loaded on →
- Is loaded on (if bulk)
- Contains ↓
- Belongs to →
- has →
- has ↓
- Leads to ↓
- Leads to

Product is mandatory, either through Item or directly linked with Piece.
Classes

Transport Means

Transport Segment

ULD

Allocated on

Is loaded on

Is loaded on (if bulk)

Contains ↓

Item

Can be in

Belongs to

Piece

Shipment

has

Waybill

has ↓

Booking

left leads to

Quotation Request

Product

Is of ↓

Can be of ↓

Product is mandatory, either through Item or directly linked with Piece

Booking

Value

ULD

VolumetricWeight

Waybill
Properties

Datatype properties

Object properties

Cardinality

Domain

Range
Turtle representation

```turtle
xsd:date rdf:type rdfs:Datatype .

# Classes
#******************************************************************************

### https://onerecord.iata.org/Address
:Address rdf:type owl:Class ;
    rdfs:subClassOf [ rdf:type owl:Restriction ;
        owl:onProperty address:addressCode ;
        owl:allValuesFrom xsd:string ;
    ] ,
    [ rdf:type owl:Restriction ;
        owl:onProperty address:addressCodeType ;
        owl:allValuesFrom xsd:string ;
    ] ,
    [ rdf:type owl:Restriction ;
        owl:onProperty address:cityCode ;
        owl:allValuesFrom xsd:string ;
```
Ontology Tools
Ontology Visualization with Widoco

IATA ONE Record Ontology

Release 30-06-2020

Latest version: https://onerecord.iata.org/
Revision: 1.0
Download serialization:
Format JSON-LD, Format RDF/OML, Format N Triples, Format TTL
License: License. License name goes here
Visualization:
OntoViz with WebGL
Evaluation:
OntoViz with OQPSO (Ontology Quality Profile Scanner)
Cite as:
IATA ONE Record Ontology. Revision 1.0.

Abstract

The ontology of IATA ONE Record standard is the core data model underlying the Linked Data solution drafted by the ONE Record standard: https://github.com/IATA-Cargo/ONE-Record. ONE Record is a standard for data sharing and creates a single record view of the shipment. This standard defines a common data model for the data that is shared via standardized and secured web API.

1. Introduction

ONE Record Data Model provides the air cargo industry with a standard data structure for data exchange using JSON-LD that facilitates data integration with existing and new data services.

1.1. Namespace declarations

<table>
<thead>
<tr>
<th>Namespace</th>
<th>Prefix</th>
<th>URI</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Ontology N8 Prefix]</td>
<td><a href="https://onerecord.iata.org/">https://onerecord.iata.org/</a></td>
<td></td>
</tr>
<tr>
<td>insurance</td>
<td><a href="https://onerecord.iata.org/Insurance">https://onerecord.iata.org/Insurance</a></td>
<td></td>
</tr>
<tr>
<td>country</td>
<td><a href="https://onerecord.iata.org/Country">https://onerecord.iata.org/Country</a></td>
<td></td>
</tr>
<tr>
<td>booking</td>
<td><a href="https://onerecord.iata.org/Booking">https://onerecord.iata.org/Booking</a></td>
<td></td>
</tr>
</tbody>
</table>
Ontology Evaluation with oops!

### IATA ONE Record Ontology

- **Title**: IATA ONE Record Ontology
- **URI**: https://onerecord.iaa.org/
- **Version**: 1.0

The following evaluation results have been generated by the RESTful web service provided by OOPS! (Ontology Pitfall Scanner).

**oops!**

It is obvious that not all the pitfalls are equally important; their impact on the ontology will depend on multiple factors. For this reason, each pitfall has an importance level attached indicating how important it is. We have identified three levels:

- **Critical**: It is crucial to correct the pitfall. Otherwise, it could affect the ontology consistency, reasoning, applicability, etc.
- **Important**: Though not critical for ontology function, it is important to correct this type of pitfall.
- **Minor**: It is not really a problem, but by correcting it we will make the ontology nicer.

### Evaluation results

<table>
<thead>
<tr>
<th>Pitfall Description</th>
<th>Detected</th>
<th>Importance</th>
</tr>
</thead>
<tbody>
<tr>
<td>P10. Missing disjointness</td>
<td>29 cases detected</td>
<td>Important</td>
</tr>
<tr>
<td>P12. Equivalent properties not explicitly declared</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P13. Inverse relationships not explicitly declared</td>
<td>127 cases detected</td>
<td>Important</td>
</tr>
<tr>
<td>P21. Using a miscellaneous class</td>
<td>3 cases detected</td>
<td>Minor</td>
</tr>
<tr>
<td>P24. Using recursive definitions</td>
<td>2 cases detected</td>
<td>Important</td>
</tr>
</tbody>
</table>

### References:

- [5] Archer, P., Goedertier, S., and Louater, N. CT 1.3 - Study on persistent URLs, with identification of best practices and recommendations on the topic for the MSs and the EC.
Object Triple Mapping with JOPA

Java OWL Persistence API (JOPA)

JOPA is a persistence API and implementation for accessing OWL ontologies. Its main features are:

- Object-ontological mapping based on integrity constraints,
- Explicit access to inferred knowledge,
- Access to unmapped properties and individual’s types
- Transactions
- Separate storage access layer

Downloads and Links

The project is hosted at [https://github.com/IATA-Cargo/one-record-server-java](https://github.com/IATA-Cargo/one-record-server-java)

JOPA artifacts can be downloaded from Maven central

JOPA implementation:

```xml
<dependency>
  <groupId>cz.cvut.kbss.jopa</groupId>
  <artifactId>jopa-impl</artifactId>
</dependency>
```

Sesame (RDF4J) OntoDriver:

```xml
<dependency>
  <groupId>cz.cvut.kbss.jopa</groupId>
  <artifactId>ontodriver-sesame</artifactId>
</dependency>
```

OWLAPI OntoDriver:

```xml
<dependency>
  <groupId>cz.cvut.kbss.jopa</groupId>
  <artifactId>ontodriver-owlapi</artifactId>
</dependency>
```
There is no single correct ontology for a domain and there is no single correct way to create it.
Deep dive into the ONE Record standard
ONE Record Insights and White Papers

Don’t miss our ONE Record Insights and White Papers

ONE Record Data Model
ONE Record & the power of ontologies
Crafting Ontologies
Object Triple Mapping
Catch the Wave of the Linked Data

https://www.iata.org/one-record/#tab-2
The first white paper explores a different way of looking at Semantic Web from the air cargo ecosystem perspective. It investigates why the usage of concepts as Linked Data and ontologies can be beneficial in a distributed end-to-end digital logistics and transport chain, such as the one that the ONE Record standard enables.
Advance your ONE Record knowledge & skills

Three great events to mark in your calendars

IATA WEBINAR
ONE Record Insights
Series of 6 webinars, every Tuesday from 23 June to 28 July, 11:00 to 12:30 CEST

SPONSORED BY
Riege Software

Digital Cargo Conference 2020
11-13 September
Week of 14-18 Sept.

All digital! All action packed! All Complimentary!

Brought to you by IATA Digital Cargo team
More info: onerecord@iata.org
ONE Record
Insights

Save the date

ONE Record Webinar
From June 23 to July 28
Every Tuesday 11:00-12:30

Hackathon
11-13 September

Q&A

Digital Cargo Conference 2020
Week of 14-18 September