## Table of Content

- One step closer to digital cargo .................................................................................. 7
- ONE Record Implementation Steps ........................................................................... 13
  - Step 1 / Define a Digital Business Process ............................................................ 16
  - Step 2 / Sign the Multilateral Data Agreement ...................................................... 19
  - Step 3 / Adopt the ONE Record Data Model ......................................................... 21
  - Step 4 / Implement the ONE Record Infrastructure ............................................... 52
  - Step 5 / Start a Pilot Project .................................................................................... 73
- Wrap up ....................................................................................................................... 74
- ONE Record progress status ..................................................................................... 77
Objective of the document

The purpose of the ONE Record Implementation Playbook is to provide a step by step guidance to implement the ONE Record standard within your organization.

The document has been designed as a pointer to the various resources made available by IATA, either the standard components or additional pieces of information/guidance materials.

This implementation playbook aims to be a catalyzer in your implementation plan.
ONE Record
One step closer to digital cargo
The Vision

An end-to-end digital logistics and transport supply chain where data is easily and transparently exchanged in a digital ecosystem of air cargo stakeholders, communities and data platforms.
The essence of the ONE Record is to move from a peer-to-peer messaging model to a data sharing model relying on a Virtual Shipment Record.
The 3 pillars

ONE Record is a standard for data sharing and creates a “Virtual Shipment Record”, i.e. a single record view of the shipment. The concept is based on 3 pillars enabling to define WHAT, HOW, with WHOM data can be shared.

The standard specifies

**Data model specification**: 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;

**API specification**: specifies the interface and interaction of the web API or Application Programming Interface that allows airlines and their partners to connect their system directly using best in class web technologies;

**Security specification**: uses an industrywide and federated trust network to manage identification and authentication of data sharing systems and ensures data privacy and confidentiality for all parties.

The standard is publicly and freely available on [GitHub](https://github.com).
The Industry benefits

The objective of ONE Record is to address the main challenges of e-freight and unlock the possibilities of a full digital air cargo industry and create opportunities for new value-added services and business models.

Data quality and control
- Data shared by data owner
- Full control of data
- Data stays at the source
- Owner determines data access

Visibility and transparency
- End-to-end transportation chain
- Share data of the shipment with relevant parties
- Enhanced visibility and transparency

Plug & Play Connectivity
- Facilitate the direct connectivity between all the stakeholders
- Use of web API
- New cooperative IT solutions and innovation

Future of digital cargo
- Foundation for true digital air cargo
- Develop collaborative and automated digital services

Welcome a new generation
- Technology platform that is ready for a new generation of digital natives

• Enhanced visibility and transparency

9
Key Points

As the industry needs to embrace change to face the current and future business and regulatory challenges, these are our conviction about what will happen:

**DIGITALIZATION**
- Complete digitalization of the global supply chain will happen
- The Internet of Logistics is a likely scenario

**AGILE SUPPLY CHAIN**
- This will lead to new and dynamic supply chain configurations
- Speed and agility is key

**REGULATORS & AUTHORITIES**
- Regulators and authorities will get high visibility and transparency
- The focus will shift to intelligence & collaboration
ONE Record
Implementation steps
The following **5 steps** are key to ensure the success of the ONE Record standard implementation:

1. Define the Digital Business Processes
2. Sign the Multilateral Data Agreement
3. Adopt the ONE Record Data Model
4. Implement the ONE Record Infrastructure
5. Start a Pilot Project
Step 1

Define the Digital Business Process
Define a Digital Business Process

While the e-freight initiative aims to digitalize a paper-based process, ONE Record enables a full data centric approach. Therefore a full digital business process can be designed, unlocking opportunities for true operations efficiency and lean organization.
Step 2

Sign the Multilateral Data Agreement
Sign the Multilateral Data Agreement

The IATA Multilateral Data Agreement (MDA) provides a single non-disclosure agreement standard whereby stakeholders sign once with IATA and start exchanging through ONE Record with all other signatory parties to the Agreement. By signing the Agreement with IATA, stakeholders effectively enter into Data Agreements with each other, i.e. enabling them to share data through the ONE Record API.

Before starting ONE Record, stakeholders are required to sign the Multilateral Data Agreement (MDA) following the below steps:

1. Complete and submit the Online Joining Form
2. Receive the Agreement by e-mail from IATA Cargo
3. Review and e-sign the Agreement
4. Receive counter-signed agreement from IATA

- Free of charge
- Provides the necessary legal framework for establishing ONE Record Data exchange
- Avoids the need to negotiate numerous bilateral Data agreements with stakeholders
- Enables to share data with all stakeholders
- Participating companies are listed on the ONE Record website

For more information, visit the [MDA page](#)
Step 3

Adopt the ONE Record Data Model
The ONE Record data model, which is one of the key pillars of ONE Record, defines what can be shared with partners.
What is the scope?
ONE Record Data Model: the ambition

- Shipper’s Letter of Instruction
- Cargo Distribution
- CO2 Emission
- ULD Tracking
- Interactive Cargo
- Dangerous Goods / Pharma
- Customs (Advance Cargo Information)
- Ground Handling

Airline Core Ontology
ONE Record Data Model: the ambition

Airline Core Ontology
• Minimal requirements for the transport of general cargo
• Detailed and extensive enough to enable piece-level management and tracking

Add-ons
• Sets of data elements required to handle specificities of certain shipments (e.g. Dangerous Goods, Pharmaceutical) and/or operations (e.g. ULD tracking, Interactive Cargo)
What is behind the data model standards?
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.

Design Principles
Conceptual Data Model
Logical Data Model
Use Cases
Ontology

The standard is publicly and freely available on GitHub.
Design Principles

The design principles document aims to describe the design principles for the data model and provide the definition of the Logistic Objects as central entities of the data model.

Definition of the four design principles

Definition of the logistic objects

Application of the data model to the Master AWB and the House AWB

What is a logistic object?

"An essential element of the cargo supply chain e.g. digital twins, transport movements, etc."
The conceptual data model describes the relationships between the Logistic Objects.
The logical data model is a detailed representation of the logistic objects and common objects. It also lists the attributes for each object, their definition and their properties (e.g. cardinality).
Use Cases

The use cases explain how the data model should be used with normal cargo operations: objects to created and modified, stakeholders involved, specificities, etc.

### ONE Record - Data Model and MOP mapping

<table>
<thead>
<tr>
<th>Action / Comment</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>YES1</td>
<td>Characteristics</td>
</tr>
<tr>
<td>YES2</td>
<td>CustomsInfo</td>
</tr>
<tr>
<td>YES3</td>
<td>Dimensions</td>
</tr>
<tr>
<td>YES4</td>
<td>Item</td>
</tr>
<tr>
<td>YES5</td>
<td>Event</td>
</tr>
<tr>
<td>YES6</td>
<td>PackagingType</td>
</tr>
<tr>
<td>YES7</td>
<td>Product</td>
</tr>
<tr>
<td>YES8</td>
<td>OtherIdentifier</td>
</tr>
<tr>
<td>YES9</td>
<td>Person</td>
</tr>
<tr>
<td>YES10</td>
<td>SecurityStatus</td>
</tr>
<tr>
<td>YES11</td>
<td>VolumetricWeight</td>
</tr>
<tr>
<td>YES12</td>
<td>ServiceRequest</td>
</tr>
<tr>
<td>YES13</td>
<td>ULD</td>
</tr>
<tr>
<td>YES14</td>
<td>YES15</td>
</tr>
<tr>
<td>YES16</td>
<td>YES17</td>
</tr>
<tr>
<td>YES18</td>
<td>YES19</td>
</tr>
<tr>
<td>YES20</td>
<td>YES21</td>
</tr>
<tr>
<td>YES22</td>
<td>YES23</td>
</tr>
<tr>
<td>YES24</td>
<td>YES25</td>
</tr>
<tr>
<td>YES26</td>
<td>YES27</td>
</tr>
<tr>
<td>YES28</td>
<td>YES29</td>
</tr>
</tbody>
</table>

The booking is made between the shipper and the forwarder, at this stage this booking is not in the scope of the data model.

The shipper ensures that the following LO are created or updated for the shipment: Product, Item, Piece, Dangerous Goods, Transport Segment (Origin and Destination, reference to the pieces), ULD if relevant (creating/updating if he is the owner, linking to existing ULD object otherwise), Security Status, Customs information, Service Request.

In this list the following are optional objects that are not mandatory at this stage: Item, ULD, Security Status, Customs information.

If there is no Item, the Product is directly linked to the Piece.

You can find the MOP Mapper [here](#).
The data model is transcribed into an Ontology (ttl file) that contains: all the objects (LO and common objects) including their description, the relationship between objects, the objects’ attributes and the cardinality.
How is it designed?
Data Model Requirements

To fulfill its purpose, the ONE Record model should meet few high-level requirements:

- Cover the end-to-end supply chain
- Find optimal balance in simplicity, flexibility and robustness of the data model
- Optimize the usage of modern technologies to facilitate data exchange
- Minimize redundancy of data in the model
Focus on the Design Principles

The ONE Record Data Model is based on four core design principles:

1. **Piece-centric**
2. **Physics-oriented: the digital twin concept**
3. **One single source of truth**
4. **Data-driven**
Piece-centric

• The Air Cargo industry is shifting from Shipment-level management to **Piece-level management** starting with Piece-level tracking

• The **Piece** is at the center of the model and deeply linked to the other elements of the cargo supply chain

What is a piece?

“A uniquely identified physical single unit which may form all or a part of a shipment”
Physics-oriented

- Physical entities have **digital twins** in the Data Model (e.g. Airplane, ULDs, etc.)
- Easy understanding of the Data Model and how it interacts with actual operations
- Easy **sharing** and **transparency** of the data throughout the supply chain

**Digital twins**

"digital twin is the "digital replica" of a physical entity"
One single source of truth

Clear ownership of data that remains at the source

Data integrity and accuracy is ensured

A strong trust is implied and in favor of replacing paper-based documents
Data, not documents!

- Data is the core of ONE Record
- Documents will be the results of data aggregation
- Proper APIs and security mechanisms allow to cover the legal requirements of documents in the current world

Combined with Semantic Web and Linked Data principles

- All objects are linked, directly or indirectly, that is Linked data
- No redundancy of data required
- The Semantic is described easily in machine-readable ontologies
How is this data model applied to the air cargo?
A model that focuses on the goods ...

Product is mandatory, either through Item or directly linked with Piece.
... has digital twins of physical assets ...

- Transport Means
- Transport Segment
- ULD
- Allocated on
- Is loaded on
- Is loaded on (if bulk)
- Contains
- Item
- Piece
- Shipment
- Can be in
- Belongs to
- Is of
- Can be of
- Product
- Product is mandatory, either through Item or directly linked with Piece
... and covers the booking process

Product is mandatory, either through Item or directly linked with Piece
## The data model through a simplified use case

<table>
<thead>
<tr>
<th>Step</th>
<th>Role</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SHIPPER</td>
<td>Prepares cargo for delivery to a customer</td>
</tr>
<tr>
<td>2</td>
<td>AIRLINE / GHA</td>
<td>Receives the shipment, arranges the booking with the airline and hands over the shipment</td>
</tr>
<tr>
<td>3</td>
<td>FORWARDER</td>
<td>At export, accepts shipment into the warehouse, loads onto a flight. At import, hands over shipment to forwarder</td>
</tr>
<tr>
<td>4</td>
<td>FORWARDER</td>
<td>Collects shipment and delivers to the consignee</td>
</tr>
<tr>
<td>5</td>
<td>CONSIGNEE</td>
<td>Consignee receives the shipment</td>
</tr>
</tbody>
</table>
Shipper prepares cargo for delivery to a customer:

- Initialization of “Piece”, “Item”, “Product” and “Shipment” objects and appropriate links
- “ULD” can be used if the shipment is already in a ULD (e.g. pharma ULD)
The data model through a simplified use case

FORWARDER

Receives the shipment, arranges the booking with the airline and hands over the shipment:

- Booking is made between forwarder and shipper, with the creation of “Quote Request”, “Offers”, “Booking” and “Waybill” objects.
- Movement of pieces to carrier domain with “Transport segments”, “Transport Means” and “Events”
- Update of the pieces’ statuses with “Events”
At export, accepts shipment into the warehouse, loads onto a flight. At import, hands over shipment to forwarder:

- Movement of pieces/ULD into the warehouses with “Transport segments”, “Transport Means” and “Events”
- Loading of pieces/ULD onto a flight with “Transport segment” and “Events”
- Departure and Arrival of the flight captured with “Events” and “Transport segment”
- Unloading of pieces from flight and loading of pieces on truck with “Transport segment”
Collects shipment and delivers to the consignee:

- Movement of pieces to the forwarder hub with “Transport segments”, “Transport Means” and “Events”
- If a ULD is broken down, the respective “ULD” and “Piece” objects are updated
- Pieces are loaded onto truck for delivery to consignee with “Transport segments”, “Transport Means” and “Events”
The data model through a simplified use case

Consignee receives the shipment:

- Delivery to consignee with "Transport segments" and "Events"
- "Event" is added to the "Piece" to reflect the final delivery
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

Browse our IATA ONE Record Resources to discover all our Insights and White Papers
Step 4
Implement the ONE Record Infrastructure
The ONE Record infrastructure is based on the three components below. Implementing these components will enable your organization to work in a ONE Record compatible environment.

- Get started with the ONE Record Ontology
- Implement the ONE Record API
- Secure the ONE Record API

The standard is publicly and freely available on GitHub.
Get started with the ONE Record Ontology
What is an Ontology?

Ontologies are frameworks for representing knowledge about concepts across a domain and the relationships between them. The ONE Record standard takes full advantage of their ability to describe relationships and their interdependence in order to model high quality, linked and coherent data.

One database may use the term “cargo”, whereas the other may use the term “freight”. To make the integration complete, an extra definition should be added to the RDF data, describing the fact that the relationship described as “freight” is the same as “cargo”. This extra piece of information is, in fact, a simple ontology.

For more information, read the ONE Record Ontology Insight
Get started with ONE Record Ontology

To incorporate the ONE Record Ontology in your systems, you can start with the below steps:

1. **Read the Data Model materials**
2. **Read the ONE Record Whitepaper**
3. **Download the ONE Record Ontology**

Catch the Wave of Linked Data with ONE Record
Looking beyond the hype for real solutions to real problems
Implement the ONE Record API
What is an API?

An API is a computing interface which defines interactions between multiple software intermediaries. It defines the kinds of calls or requests that can be made, how to make them, the data formats that should be used and the conventions to follow.

For more information, read the [ONE Record API Insight](#)
Implement the ONE Record API

There are two ways to implement the ONE Record API

Code your own ONE Record API from scratch by using the ONE Record API Specifications and the ONE Record Ontology

Download the ONE Record Sandbox and integrate it into your system

Download the API Ontology

Download the Java Sandbox

Read the API specifications
What are the main features of the ONE Record API?
Introducing Tom & Jerry, API friends

Hi my name is Tom, I am the data owner

Hi my name is Jerry, I am the data consumer

Together we will walk you through the API features
Let’s hear from Tom & Jerry challenges

Tom & Jerry are going to walk us through the different API features

1. How do I make my data available?
2. How can I access the data?
3. How do I raise a change request?
4. How do I update the data?
5. How can I save the history of the data?
6. How do I give data access to my partners?
7. How can we automate data notifications?
8. How can I send events related to data?
9. How can I define to whom I give data access?
10. How can I take a snapshot of the data?
11. How can I retrieve a version of data at a certain moment in time?
12. How can I see all the existing versions of the data?
How do I make my data available?

Publishing data with POST

When creating a new Logistics Object (LO) on a ONE Record Server, you need to do a HTTPS POST request. The data for the LO should be included in the request body and provided that you are authenticated and authorized, the server will accept the request and create a new LO. This operation will be generally performed by the owner of the data, who in most cases owns or at least controls the server.
How can I access the data?

**Reading data with GET**

To read the content of a Logistics Object, you need to perform a HTTPS GET request. The server that you are accessing will check that you are an authenticated and authorized user before it will return you the data. JSON-LD (application/ld+json) is the standard response format for the ONE Record API.
How do I raise a change request?

Change request with PATCH

Whenever you need to request a change to data in a Logistics Object, you need to use the HTTPS PATCH method. In ONE Record API, the PATCH request represents an array of objects. Each object represents a single operation to be applied to the target Logistics Object (add and/or delete).

For more information, read the ONE Record PATCH Insight.
How do I update the data?

**Updating data with PATCH**

Only the publisher can change the Logistics Object, where the publisher is the party that creates the Logistics Object on the ONE Record server. The evaluation of a PATCH request occurs as a single event. Operations are sorted and processed as groups of delete and then add operations until all the operations are applied, or the entire PATCH fails.

The example below describes the change to be made – delete the `totalPieceAndULDCount` of value 10 and add value 11 instead. Also, a new field – `date` – is added.

For more information, read the [ONE Record PATCH Insight](#).
How can I save the history of the data?

Audit trail of the changes

An audit trail (history) of all the change requests is stored and can be retrieved at any moment from a dedicated endpoint on the ONE Record API.

```json
*create*:
  *lo*:"initial content of the Logistics Object"
},
*logisticsObjectRef*:"Logistics Object Id to which the audit trail applies",
*changeRequests*:[
  { "timestamp":2019-09-17T14:49:13+00:00",
  "companyId":"http://myonerecordserver.com/AIRLINE",
  "changeRequest":{
    "revision":1",
    "description":"Updated number of total pieces count",
    "operations":[
      { "op":"del",
        "p":"http://onerecord.iata.org/Waybill#totalPieceAndULDCount",
        "o":{
          "value":10",
          "datatype":https://www.w3.org/2001/XMLSchema#decimal" },
      { "op":"add",
        "p":"http://onerecord.iata.org/Waybill#totalPieceAndULDCount",
        "o":{
          "value":11",
          "datatype":https://www.w3.org/2001/XMLSchema#decimal" },
      { "op":"add",
        "p":http://onerecord.iata.org/Waybill#date",
        "o":{
          "value":2019-08-18",
          "datatype":https://www.w3.org/2001/XMLSchema#decimal" }}
    ]
  }
},
"status":"ACCEPTED"
]}
```
How do I give partners access to my data? (1/2)

Access Delegation

Typically, the company that has created the data will notify their partner and provide them access details such as the URI of the data. However, that second company may need to share the same data with another company downstream. This can be performed via the access delegation feature.
Chains of trust are based on business partnerships and trust in the transport chain. It ensures that the company who has shared a logistics object on a server, always knows who may access this and at any time, it can revoke all or part of the chain of trust.

For more information, read the Access Delegation Insight.
How can we automate data notifications?

**Automatic data updates through pub/sub**

In distributed applications, components of the system often need to provide information to other components as events happen. For example, companies need to be notified when new data becomes available, so they can act accordingly if required.

For more information, read the [Publish & Subscribe Insight](#).
How can I send events related to data?

Status updates via Events

Status updates in ONE Record can be added to Logistics Objects through Events. By definition, each Logistics Object can be assigned Events.

POST http://localhost:8080/companies/myCompany/los/AWB-445555566/events
How do I define to whom I give access to the data?

Access Control Lists

In ONE Record, access to resources can be handled by using Access Control Lists (ACLs) stored in the backend systems of the ONE Record Servers and defined using the Web Access Control standard from W3C. Each Logistics Object resource has a set of Authorization statements describing who has access to that resource and what types (or modes) of access they have.

READ / GET
Read the contents (including querying it)

WRITE / POST and PATCH
Write contents or modify part of it

CONTROL
Read and Write
How can I take a snapshot of the data?

**Memento**

A Web resource that is a prior version of the Original Resource, i.e. that encapsulates what the Original Resource was like at some time in the past. In ONE Record, a Memento contains a snapshot of the data at a certain moment in time.
How can I retrieve a version of data at a certain moment in time?

**TimeGate**

A Web resource that “decides” on the basis of a given datetime, which Memento best matches what the Original Resource was like around that given datetime. When negotiating with the TimeGate, the client uses an `Accept-Datetime` header to express the desired datetime of a prior/archived version of the original resource. The TimeGate responds with the location of a matching version, a Memento.
How can I see all the existing versions of the data?

**TimeMap**

A TimeMap is a machine-readable document that lists the Original Resource itself, its TimeGate, and its Mementos as well as associated metadata such as archival datetime for Mementos. TimeMaps are exposed by systems that host prior versions of Original Resources and allow for batch discovery of Mementos.
Secure the ONE Record API
Secure the ONE Record API

In order to meet the ONE Record security requirements, the security architecture is based on two layers:

1. **ONE Record Federated Security Agreement**
   - Certificate Authority Compliance
   - Certificate Profile
   - Registration procedures
   - Service Level Agreements

2. **Certificate Authority Services and Verification Services**
   - Signed ONE Record Certificates
   - ID accounts + tokens

3. **Identity & Authentication Service Provider**

4. **Single Node**
   - User: Password:

5. **Multi-company Node**
   - User: Password:

6. **Multi-user Node**
   - User: Password:

7. **Multi-device Node**
   - User: Password:
MUTUAL TLS

Mutual TLS (HTTPS)
- Mutual TLS secures the Node to Node channels
- Ensures PKI encrypted data channel
- Ensures that only ONE Record recognized servers are used

Certificate Authorities (CA)
- Issue and authenticate valid ONE Record certificates
- Must be internationally accredited to issue public certificates
- Meets ONE Record requirements for registration and service levels
- Is federated with other certificate authorities and identity & authentication services
OAuth 2.0 and Open ID Connect

- Framework for identification and authentication
- Open ID Connect facilitates user-based accounts
- Ensures identity & authenticity of users, companies and devices
- Ensures identity & authenticity of the IAP that issues the token for data exchanges
- Provides signed & encoded encryption of token (JWT) and payload

Identity & Authentication Providers (IAP)

- Meets ONE Record requirements for registration and service levels
- Is federated with other IAP’s and CAs
- Can register companies, users and devices and provide user accounts
- Must hold a signed certificate issued by a ONE Record Certificate Authority
- Authenticates users and issues token for secured data exchange
Deep dive into the ONE Record Security

To know more about the implementation of the ONE Record Security mechanisms, please check our dedicated webinar on Security.
Step 5

Start a Pilot Project
The objectives of the pilot projects are to verify that the ONE Record standard components are fit for purpose, i.e. they bring the expected value, and to capture the lessons learned as an input for standard improvement. To join our pilot project program, please follow the below steps.

**Process**
1. Formalize a specific use case
2. Start a pilot project
3. Report Pilot findings to IATA

**Existing Uses Cases**
- Cargo distribution
- Cargo pre-advice for trucking
- Customer integration
- Enriching commercial documents
- ONE Record Airport hub
- ONE Record and Existing Messaging Standards
- ONE Record node expansion
- ONE Record prototyping
- Pharmaceuticals
- ONE Record for Road Transport

For more information, visit our ONE Record Pilot Project [webpage](#)
WRAP UP
## ONE Record Implementation Steps

### 1 Define the Digital Business Processes
- Identify the value-added activities
- Map the information flow against the physical flow
- Define the Target Operating Model
- Design the associated digital business processes

### 2 Sign the Multilateral Data Agreement
- Complete and submit the [Online Joining Form](#)
- Receive the Agreement by e-mail from IATA Cargo
- Review and e-sign the Agreement
- Receive counter-signed agreement from IATA

### 3 Adopt the ONE Record Data Model
- Read the Design Principles
- Integrate the Logical Data Model in your systems
- Understand the use cases

### 4 Implement the ONE Record Infrastructure
- Get started with the ONE Record Ontology
- Implement the ONE Record API
- Secure the ONE Record API

### 5 Start a Pilot Project
- Formalize a specific use case
- Start a pilot project
- Report Pilot findings to IATA

---

For more information reach us at [onerecord@iata.org](mailto:onerecord@iata.org)
ONE Record
Progress Status
In order to monitor the progress of the ONE Record standard development, IATA published a standard development dashboard and a quarterly newsletter.

ONE Record Progress Status

ONE Record Quarterly, a quarterly newsletter which will give you an update on the ONE Record development and engagement activities.
Check our latest list of Pilot Project participants [here](#).
Thank You

More info

www.iata.org/one-record