OMMT
UIC Open Multi-Modal Toolkit

Technical Guidelines v1.0

UIC - International Union of Railways

Contents

List of figures 3
List of Abbreviations 4
Executive Summary 5
1. Introduction 6
2. Market Context 6
3. Reference Models 8
3.1. Multimodal Partnership Models 8
3.2. Integrated Mobility Experience Models 9
4. Overview of OMMT 10
4.1. Goals 10
4.2. Audience 10
4.3. Benefits 11
4.4. Components 12
4.5. Applicability to Multimodal Partnership Models 15
4.6. Applicability to Integrated Mobility Experience Models 19
5. OMMT.Plan Overview 20
6. OMMT.Book Overview 20
6.1. OSDM main concepts 20
6.2. OSDM basic booking process flow 22
6.3. OSDM essential data models 22
6.4. Available resources 28
7. OMMT.Move Overview 29
7.1. Conversion of data elements into barcodes 29
7.2. Data composition 29
7.3. Signing and verifying the data 30
7.4. Flexible Content Barcode (FCB) 31
7.5. Dynamic Flexible Content Barcode - DOSIPAS 34
7.6. Key management 36
7.7. Available resources 37
8. OMMT.Sync Overview 37
8.1. eTCD process and functions 38
8.2. Available resources 40
9. OMMT.Share Overview 41
9.1. Available resources 43
10. Quick guides 43
10.1. Quick guide for Orchestrators 44
10.2. Quick guide for Scheduled Transport Operators (bus, metro, rail, ferries) 44
10.3. Quick guide for Bike / Scooter / Car Sharing Operators 47
10.4. Quick guide for Taxi / Ride-hailing / Demand Responsive Transport Operators 49

10.5. Quick guide for Parking Operators 51
11. Conclusions 54

List of figures

Figure 1 - MaaS models according to UITP ____ *8*
Figure 2 - The five Integrated Experience models ____ *9*
Figure 3 - OMMT components ____ *13*
Figure 4 - SP with Orchestrator role ____ *16*
Figure 5 - Pure Orchestrator ____ *17*
Figure 6 - Retailer separate from Orchestrator ____ *17*
Figure 7 - Multiple Orchestrator levels ____ *18*
Figure 8 - Multiple Orchestrators ____ *18*
Figure 9 - Interactions enabled by OSDM ____ *21*
Figure 10 - Place data model ____ *23*
Figure 11 - Trips data model ____ *24*
Figure 12 - Offer resource attributes ____ *25*
Figure 13 - Signing and verification with private and public key pair ____ *31*
Figure 14 - Generating a Dynamic Flexible Content Barcode ____ *35*
Figure 15 - Verifying a Dynamic Flexible Content Barcode ____ *35*
Figure 16 - Keypair validity example ____ *37*
Figure 17 - Actors and Interactions in eTCD ____ *38*
Figure 18 - Almost real-time sharing of annotations enabled by eTCD ____ *39*
Figure 19 - NGRS model for Pre-purchased ticket – commission ____ *41*
Figure 20 - NGRS model for Pre-purchased pass – pro-rata ____ *42*
Figure 21 - NGRS model for Pre-purchased pass – usage ____ *42*
Figure 22 - NGRS model for Account-based model ____ *43*

List of Abbreviations

Abbreviation	*Meaning*
API	Application Program Interface
D2D	Door-to-Door
DB	Deutsche Bahn
DOSIPAS	Double Signed Package Structure
DRT	Demand Responsive Transport
EMV	Europay, Mastercard and Visa
eTCD	E-Ticket Control Database
EU	European Union
FCB	Flexible Content Barcode
GTFS	General Transit Feed Specification
HSM	Hardware Security Model
IANA	Internet Assigned Numbers Authority
IM	Infrastructure Manager
IT	Information Technology
JSON	JavaScript Object Notation
MaaS	Mobility as a Service
MERITS	Multiple East-West Railways Integrated Timetable Storage
MSP	Mobility Service Provider
NFC	Near Field Communication
NGRS	Next generation Revenue Sharing
OJP	Open Journey Planner
OSDM	Open Sales and Distribution Model
PKMW	Public Key Management Website
SDK	Software Development kit
SiD	Security in Data
SiS	Security in System
TCN	Ticket Control Number
TCO	Ticket Controlling Organizations
TSGA	Transport Service Governance Authority
UIC	International Union of Railways
UITP	International Association of Public Transport

Executive Summary

This report presents technical guidelines for implementing and operating OMMT, a unique UIC solution designed to meet the needs of railways and their partners involved in multimodal offerings.

Achieving interoperability among systems and processes of multiple transportation modes and mobility operators is one of the main challenges for the development of integrated mobility. Digital technologies reduce the dependencies from legacy systems, offer more flexibility, allow fully dematerialised user mobility experiences and open new opportunities. OMMT provides a set of open digital specifications and services that simplify the implementation and operation of key components of multimodal door-to-door mobility, including booking, validation, inspection, and revenue settlement among partners. It is versatile and flexible. As described in this report, it supports multiple MaaS models and enables door-to-door mobility user experiences that may include reservations, pre-purchasing of digital integrated or non-integrated tickets, as well as pay-as-you-go and account-based mobility. It enables international partnerships and cross-border travel.

Far from being a monolith, the architecture of OMMT is modular, which makes possible the adoption of individual components or of the full solution. It is intended to cooperate with other solutions and industry standards.

OMMT is grounded on UIC specifications and solutions that have already been adopted internationally by rail undertakings and other transport providers. This report presents in an easy-to-read form the essential elements that are needed to implement multimodal door-to-door solutions. It also includes quick guides tailored for each type of mobility provider, allowing, for example, a public transport operator or a bike sharing provider to easily understand how to adopt OMMT.

Mobility authorities and transportation industry players should look at OMMT as a resource that railways can provide, in a cooperative fashion, as a contribution to facilitate and accelerate the implementation and delivery of integrated mobility to citizens.

1. Introduction

OMMT (UIC Open Multi-Modal Toolkit) is a one-stop solution serving the needs of railways and their partners involved in multimodal offerings**.** This report presents the OMMT concept and the first version of OMMT Technical Guidelines. It is intended not only for railways, but also for partners and stakeholders (PTAs, PTOs, other transportation service providers, MaaS providers) involved in the implementation of digital integrated mobility services in cooperation with railways.

Here follows an **outline of this report**:

- Chapter 2 provides a summary of the market context
- Chapter 3 introduces reference models for multimodal partnerships and for delivering integrated mobility experiences
- Chapter 4 provides an overview of OMMT
- Chapters 5 to 9 describe the five pillars of OMMT (**OMMT.Plan**, **OMMT.Book**, **OMMT.Move**, **OMMT.Sync**, and **OMMT.Share**)
- Chapter 10 contains quick guides on how to implement OMMT, customised for each type of player in multimodal mobility
- Chapter 11 provides conclusions

2. Market Context

An unprecedented **transformation is taking place in mobility**, powered by digital technologies. Key drivers of the transformation include:

- New mobility services, such as ridesharing and vehicle-sharing
- The push towards sustainable and inclusive mobility
- Innovative paradigms, like Mobility-as-a-Service and Demand Responsive Transport
- Users' raised expectations of more user-centric and seamless services
- The evolution (or, in some cases, disruption) of traditional approaches to ticketing

As far as ticketing is concerned, a key trend is the **demateralisation** of transportation contracts. Increasingly, the proof of entitlement to travel and any records of travel are held in a central system and not on physical media held by the passenger, such as a paper ticket or a smart card. To prove their entitlement to travel, passengers need a unique identifier linked to their e-ticket or account. This can be a cEMV payment card, an identifier communicated via smartphone (via display, NFC or Bluetooth), or a home-printed barcode.

Dematerialisation regards **all mobility modes**. The airline sector has abandoned paper tickets since 2008. Increasingly, rail passengers are traveling with e-tickets. Account-based ticketing is being adopted by a growing number of public transport organisations and by public transport authorities for integrated multimodal schemes. Mobility services like ridesharing and vehicle-sharing are natively digital and account-based.

Traditional integrated multimodal offerings are often limited to a region (or country) and managed by a single Transport Authority. **Dematerialisation reduces the dependencies from legacy equipment, offers more flexibility and opens new opportunities.** Rail undertakings and other transport service providers that operate across regions and borders can create more flexible partnerships and offerings to customers.

Roaming is a common concept in mobile communications. Mobile users, when traveling, can easily access services by different telco providers. They do not need a different contract or SIM for every country they visit. Standardisation and interoperability enable telco providers to establish flexible partnerships and to provide an integrated, seamless experience to customers. Interoperability in travel and mobility services is definitely more complex, as multiple modes and numerous providers are involved. However, creating seamless “roaming” experiences for travelers across regions and borders is an objective that can be achieved or at least approached by leveraging available technologies and technical specifications.

3. Reference Models

There are many ways to provide digitally integrated mobility solutions. OMMT does not embrace a single model. It aims at being flexible and versatile in order to adapt to different models. This chapter introduces the main **multimodal partnership models** and **integrated mobility experience models** that are available to provide multimodal mobility. The next chapters will explain how OMMT can be applied to the different models.

3.1. Multimodal Partnership Models

In its 2019 Policy Brief *Ready for MaaS?,* UITP, the International Association of Public Transport, introduces 4 Mobility-as-a-Service (MaaS) models, summarised in the following figure with some representative examples.

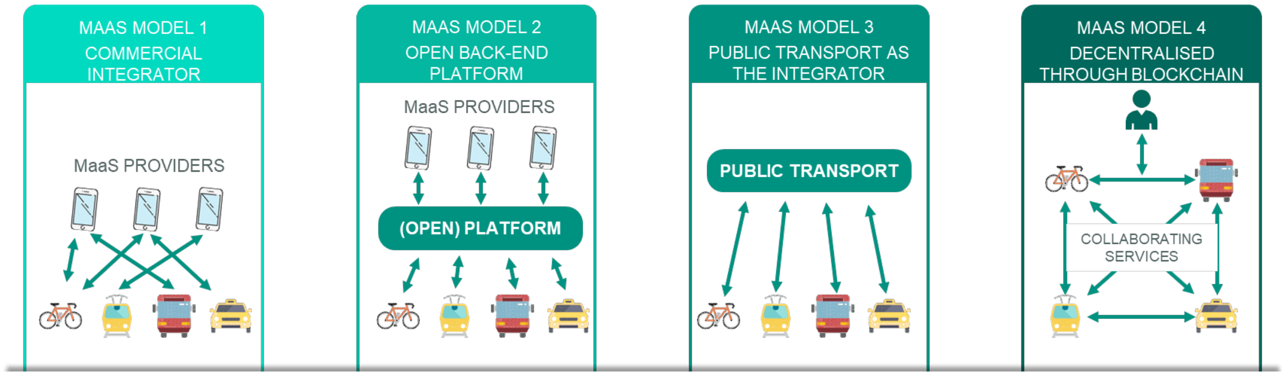

Figure 1 - MaaS models according to UITP

The key characteristics of each model can be outlined as follows:

- **Model 1 (Commercial Integrator)** is based on a marketplace with agreements between MaaS providers and transport operators. It promotes competition among multiple MaaS providers.
- **Model 2 (Open Back-end Platform)** relies on a common integration platform, that typically is set up by a public entity with rules determined by the public authority. It serves as a public infrastructure on which different actors can build a MaaS solution**.**
- **Model 3 (Public Transport as the Integrator)** assumes that a public transport entity runs MaaS with selected mobility services.
- **Model 4 (Decentralised)** allows all stakeholders to work on a shared platform via distributed ledger technology. It prevents the need for a single entity in the middle, while still creating a coordinated mobility ecosystem. Research on this possibility is still ongoing.

The above concepts can be generalised and applied to essentially any multimodal partnership, not necessarily labelled as “MaaS”. By “multimodal partnership” we mean a collaboration among entities that enter into commercial agreements to jointly sell and deliver mobility services across multiple transport modes and share the resulting revenues.

In a multimodal partnership, each entity can play one or more of the following roles: Service Provider (supplier of passenger transportation and related services), Orchestrator (combining services of multiple Service

Providers to enable an integrated experience for the customer), and Retailer (interfacing the customer and collecting payments). The Orchestrator can be a pure integrator, like in UITP models 1 and 2, or a Service Provider, like in UITP model 3. The Retailer can be the same as the Orchestrator, like in UITP models 1 and 3, or separate, like in UITP model 2. Multiple Orchestrators are possible, like in UITP model 4 (not necessarily enabled by blockchain technology).

3.2. Integrated Mobility Experience Models

There are different ways to enable an integrated passenger experience, encompassing multiple mobility providers and modes. Based on an analysis of case histories and best practice, the UIC D2D Implementation project has identified five main models.

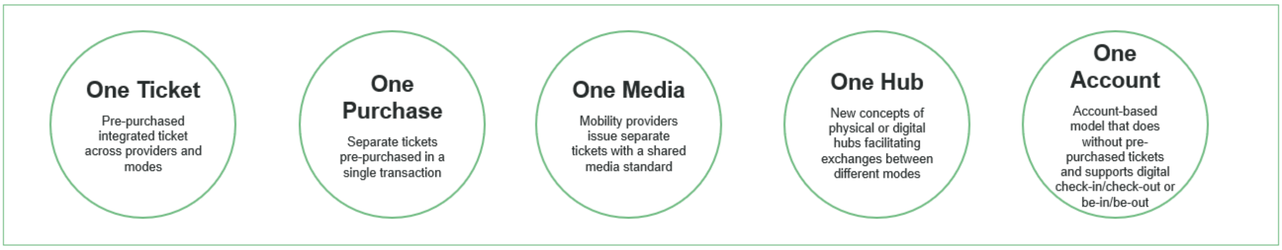

Figure 2 - The five Integrated Experience models

An outline of the five models follows.

- **One Ticket:** Pre-purchased integrated ticket across providers and modes

  Customers buy in advance a single ticket that can be used to travel with different providers and modes

- **One Purchase:** Separate tickets pre-purchased in a single transaction

  Customers buy in advance, in a single transaction, separate tickets and use them to travel with different providers and modes.

- **One Media:** Mobility providers issue separate tickets with a shared media standard

  Mobility providers issue separate tickets with a shared media standard, for example a barcode, that is used as a fulfillment for allowing the passenger to access the mobility services.

- **One Hub:** New concepts of physical or digital hubs facilitating exchange between different modes

  This model is not directly linked with ticketing aspects, but it refers to the creation of physical or digital hubs that facilitate the exchange between different modes; especially useful when sharing mobility is involved.

- **One Account:** Account-based model that does without pre-purchased tickets and supports digital check-in/check-out, tap-in/tap-out or be-in/be-out

  Customers do not need to pre-purchase tickets. Mobility services are charged to their account, that can be either a general payment account (open-loop) or an account dedicated to mobility services (closed-loop).

4. Overview of OMMT

4.1. Goals

OMMT, the UIC Open Multi-Modal Toolkit, aims at providing a one-stop solution serving the needs of railways and their partners involved in multimodal offerings.

OMMT goals include:

- Support multiple multimodal partnership models.
- Support multiple integrated mobility experience models.
- Minimise the effort required to railways, once they are aligned with recent UIC specifications.
- Minimise the effort required to railways' multimodal partners.
- Simplify the design and implementation of multimodal offerings through modular, integrated, standardised, scalable components.
- Facilitate the adoption by the mobility ecosystem through open specifications and simple implementation guidelines.

4.2. Audience

This Technical Guidelines document is open to all parties who may be interested in OMMT, including:

- **RUs**

  Rail Undertakings that intend to play a leading role in the development of integrated mobility and/or plan to participate in multiple integrated mobility/MaaS schemes.

- **PTAs**

  Public Transport Authorities and other agencies that would like to implement integrated mobility service policies to improve citizens' mobility and promote sustainable behaviours.

- **PTOs**

  Public Transport Organizations that intend to implement digital multimodal transport services by taking advantage of collaborative opportunities.

- **Other Mobility Service Providers**

  All other mobility service providers (buses, ferries, ridesharing, vehicle sharing, parking, ride-hailing, etc) that plan to participate in multimodal mobility partnerships.

- **MaaS Providers**

  MaaS Providers who face the challenges of diverse and fragmented models and interfaces when integrating multiple modes and providers.

4.3. Benefits

OMMT can unlock benefits for all parties involved.

Rail Undertakings

The benefits that OMMT enables for Rail Undertakings are the following:

- **Enhanced interoperability:** integration with other services and partners are facilitated and improved, enabling railway companies to offer multimodal services more easily.
- **Common industry specifications:** confidence in investing in technical solutions, by RUs and their partners, is enhanced by international industry specifications.
- **Optimised implementation effort:** railways can implement new functionalities or services quickly and efficiently and can leverage common industry services.
- **Solutions reusable in multiple contexts:** railways are facilitated in replicating solutions in multiple markets, scaling their efforts, onboarding new partners and adapting to changes in the market.

PTAs and other authorities

PTAs and other authorities in charge of mobility policies can benefit from OMMT in several ways:

- **Facilitation and acceleration of MaaS initiatives:** collaboration among industry players to offer integrated mobility services can be accelerated by adopting a ready framework of common international technical specifications.
- **Improved mobility for citizens:** PTAs can adopt the MaaS model that can best improve the mobility experience of citizens and visitors, leveraging the versatility of OMMT.
- **Enablement of policies for sustainable mobility:** policies that incent users to make sustainable choices, such as using rail in conjunction with green first- and last-mile mobility, are enabled.

PTOs

The benefits of OMMT for PTOs are the following:

- **Collaboration Opportunities:** PTOs are facilitated when collaborating with railways and other transportation stakeholders to offer integrated solutions.
- **Plug and play solutions for implementing smart digital mobility:** PTOs can quickly integrate new digital services into their existing operations.
- **Support of Modal Shift:** Offering convenient mobility solutions integrated with rail can encourage modal shift from individual modes of transportation.

Other Mobility Service Providers

All types of Mobility Service Providers can benefit:

- **Access to Broader Market:** Mobility Service Providers have the opportunity to access a wider market, unleashing business opportunities, diversification of offering, and higher revenues.
- **Simplified Integration:** OMMT enables reduced implementation cost and time, making at the same advanced functionalities accessible.
- **Business Growth Opportunities:** Mobility Service Providers have the possibility to diversify their business, expand the portfolio of activities and develop strategic collaborations.

MaaS Providers

The benefits of OMMT for MaaS Providers are the following:

- **Plug-and-Play Integration:** MaaS Providers will be able to quickly integrate new digital services into their existing operations.
- **Expanded Service Offerings:** through OMMT, MaaS providers will be able to offer a wider range of services, thereby increasing customers' flexibility and choice in planning their trips.
- **Solutions reusable in multiple contexts:** MaaS Providers will be able to reuse solutions in multiple markets, as OMMT relies on non-proprietary, international specifications.

Travellers

Last, but not least, travellers can benefit from OMMT in different ways:

- **Seamless Journeys and Customer Experiences:** seamless travel experiences are enabled when multiple modes or service providers are involved.
- **Simplified Ticketing:** a simplified digital ticketing process is enabled, enhancing the experience of buying and using tickets.
- **More convenient choices:** development of new mobility offerings is facilitated, providing more choice to customers.

4.4. Components

OMMT assumes that a **commercial agreement** has been established among Service Providers to offer integrated mobile services to customers. The modular and open nature of OMMT allows to easily implement such agreements, as well as any updates, for example adding new participating Service Providers.

OMMT includes **five components**:

1. **OMMT.Plan**

2. **OMMT.Book**
3. **OMMT.Move**
4. **OMMT.Sync**
5. **OMMT.Share**

They are **modular** (ie each one can work independently of the others) and **integrated** (ie they work nicely together if more than one is adopted).

The five components of OMMT provide solutions for the key capabilities required for providing digital door-to-door mobility services, as summarised in the following figure.

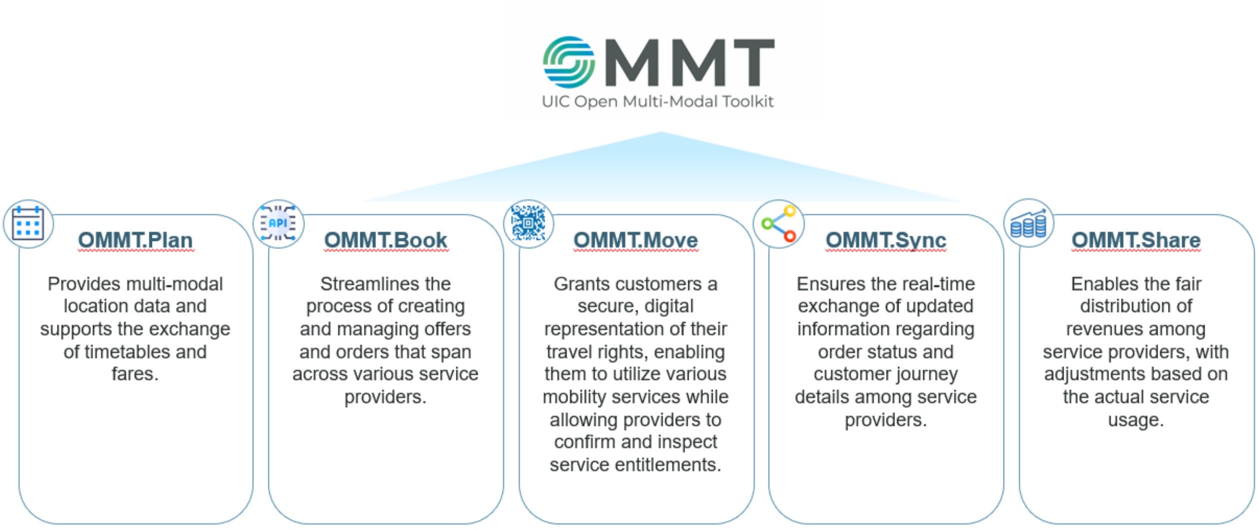

Figure 3 - OMMT components

A brief description of each component follows.

OMMT.Plan

OMMT.Plan provides a service for the **exchange of timetable data**, including in GTFS format, a **fare distribution** system, and a **location API** the allows to search for location codes (to be used in the booking step to indicate origin and/or destination of a trip) based on a string describing a place or by its geocoordinates.

OMMT.Book

OMMT.Book allows providers to exchange offers, that may include dynamic prices, and book services, through an **open-source API**.

One of the providers acts as a **distributor** that assembles and sells services to a customer. The distributor, for example, can be a rail undertaking that offers to customers an integrated ticket (or multiple tickets in a single purchase transaction) combining rail transport services, in connection with parking, long distance bus and public transport services. Or the distributor can be a MaaS provider that assembles a mobility package that includes a pass to public transport.

The distributor may provide a **common digital representation** of the rights acquired by the customer, that allows the customer to access the different mobility services, through **OMMT.Move**. Or, through the **OMMT.Book** API, **each provider may supply its own digital encoding of the rights** acquired by the customer: for example, a parking operator can provide a numerical code to be input for entering the parking and a PTO can provide a QR code recognized by its own validators and inspectors. OMMT is agnostic about how the providers should share network and timetable information among them and about the journey planning step. If the distributor offers a journey planning service to the customer, the output of the journey planner can be fed into **OMMT.Book** for booking. **OMMT.Book** is **natively interoperable with the Open Journey Planner (OJP) standard**.

OMMT.Move

OMMT.Move allows customers to move around by easily accessing the mobility services to which they are entitled. It includes the **specifications** for encoding the acquired mobility services (or the account held by the customer in case of account-based mobility) into a **flexible content, secure 2D-barcode** that customers can carry with their smartphone. It also provides the specifications for mobility providers to decode and verify the 2D-barcode, either automatically (e.g., at a metro station gate) or manually (e.g., by an inspector onboard a bus). The specifications come with a **software implementation that providers can quickly incorporate** in their own applications.

For additional security, **OMMT.Move** also supports dual-signature **dynamic barcodes**, through the DOSIPAS solution (see dedicated section describing DOSIPAS). The PKMW service supports the **distribution of public keys**.

OMMT.Move can also provide a standard layout to inform the passenger about key itinerary and contractual information.

OMMT.Sync

OMMT.Sync enables **sharing in real-time** updated information about the status of the order and of the customer journey among Service Providers, when needed.

Some examples follow. When a pass is issued to a customer, multiple mobility providers may need to be made aware of it. When the customer has an open entitlement to a single bus ride, the information that the customer has consumed the ride should be shared to avoid that the customer tries to use the ride twice. In account-based travel, sharing information on events (such as the customer has been issued an account, or the customer has performed a check-in with a mobility service) is critical and is needed for charging the customer and sharing revenues among providers.

OMMT.Sync is based on an **API specification and a centralized data sharing service**, that providers can easily integrate with their own IT applications.

A future add-on to **OMMT.Sync** is a central service allowing to share in real-time data about **delays** and **expected time of arrival**.

OMMT.Share

OMMT.Share enables **apportionment of revenues among providers**, based on the commercial agreement. It supports multiple models, including commission based, settlement value, fixed apportionment, and dynamic apportionment based on actual usage. No IT integration effort is required by providers, that will receive a report with settlement amounts and a detailed transaction log**.**

4.5. Applicability to Multimodal Partnership Models

The components of OMMT can be applied in different ways to enable different Multimodal Partnership Models. Some examples follow.

SP with Orchestrator role

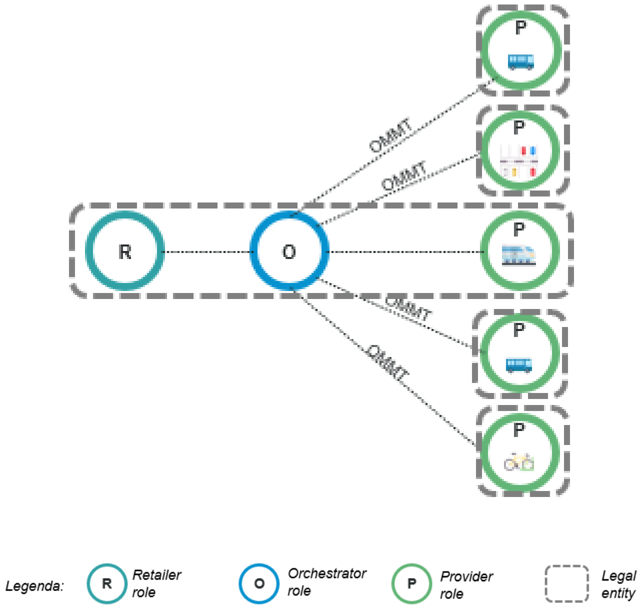

Figure 4 - SP with Orchestrator role

The Orchestrator can be a Service Provider. For example, a rail operator that through OMMT offers an integrated experience combining rail transportation with parking and local public transport, will have the double role of provider of rail transportation services and Orchestrator. The Orchestrator can act as Retailer as well.

Pure Orchestrator

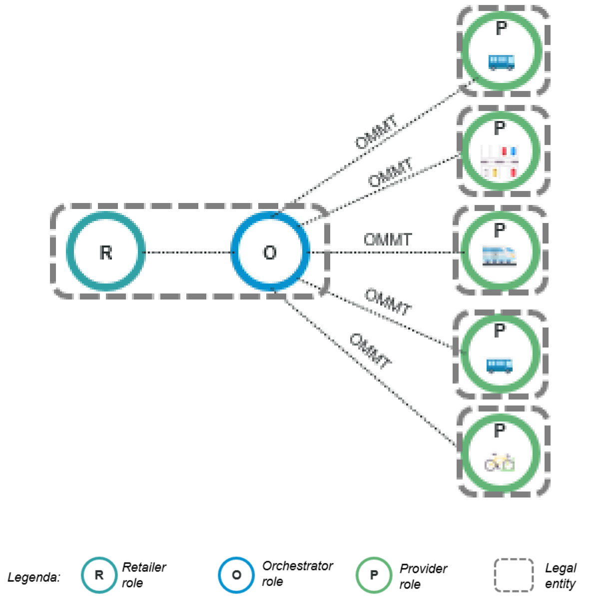

Figure 5 - Pure Orchestrator

One company, for example a pure MaaS Integrator, could act as Orchestrator without being at the same time a Provider.

Retailer separate from Orchestrator

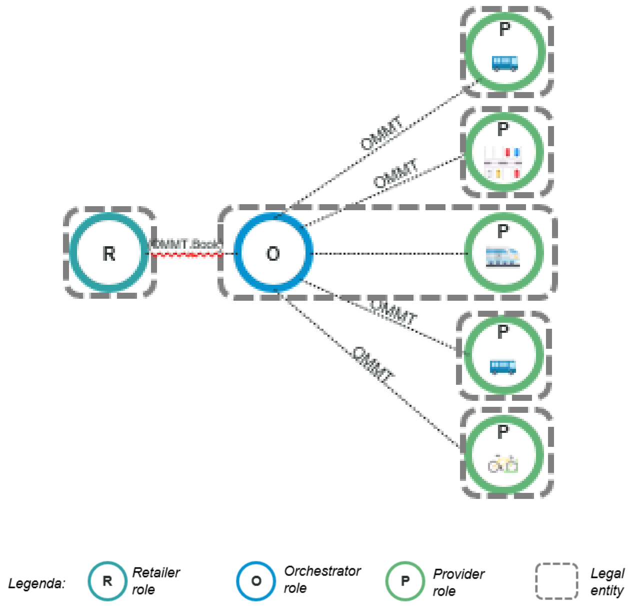

Figure 6 - Retailer separate from Orchestrator

The Retailer can be a separate company that interfaces the Orchestrator through **OMMT.Book** or a different API.

Multiple Orchestrator Levels

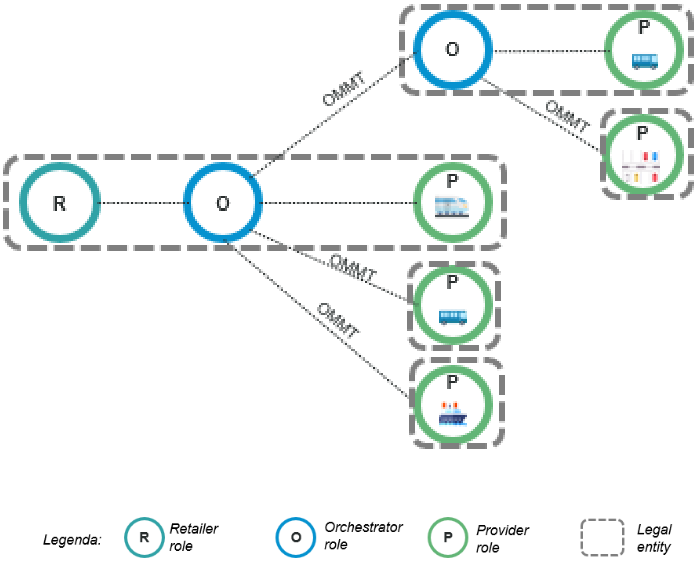

Figure 7 - Multiple Orchestrator levels

Multiple levels of orchestration are possible, i.e. an Orchestrator can act as supplier of another Orchestrator.

Multiple Orchestrators

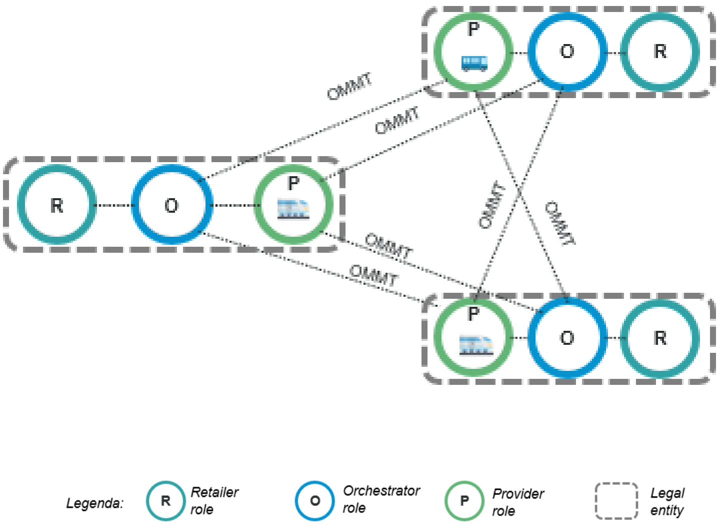

Figure 8 - Multiple Orchestrators

Providers can partner and agree that each can act as Orchestrator of services of the other Providers.

4.6. Applicability to Integrated Mobility Experience Models

The components of OMMT can be applied in different ways to enable each of the five reference Integrated Mobility Experience Models introduced previously.

- In the **One Ticket** model, **OMMT.Book** can be the interface between retailer and suppliers that enables integrated ticketing (and booking if applicable), **OMMT.Move** can be used to create and control the digital fulfilment of the integrated ticket, **OMMT.Sync** allows secure, real-time sharing of the ticket status among partners, **OMMT.Share** simplifies the sharing and accounting of ticket revenues among contributing service providers.
- In the **One Purchase** model, **OMMT.Book** can be the interface between retailer and suppliers that enables issuing separate tickets (and bookings if applicable), **OMMT.Move** can be used to create and control the digital fulfilment of tickets, **OMMT.Sync** allows secure, real-time sharing of the ticket status among partners, **OMMT.Share** simplifies the sharing and accounting of ticket revenues between the retailer and the suppliers.
- In the **One Media** model, **OMMT.Move** can be used to provide common media for the different tickets, and **OMMT.Sync** supports secure validation and control of the barcodes.
- In the **One Hub** model, **OMMT.Book** can be used to share information about the availability of vehicles / services at hubs and to support the booking.
- In the **One Account** model, **OMMT.Book** can be the interface between the account provider and the suppliers, **OMMT.Move** can be used to create and control the digital identification of the account and, when applicable, to represent check-in/check-out status, **OMMT.Sync** allows secure, real-time sharing of valid accounts and of check-in/check-out data, **OMMT.Share** simplifies the accounting and charging of post-paid services to customers, as well as the settlement of revenues between the account provider and the service providers.

It is worth underlining that when a specific component of OMMT is applicable to a model, it is not mandatory to apply it. For example, in the One Account model the account id is not necessarily represented by a barcode, which makes **OMMT.Move** optional.

In addition, **OMMT.Plan** can be useful to support the Journey Planning step in any of the Integrated Mobility Experience Models.

5. OMMT.Plan Overview

Solutions for door-to-door journey planning are widely available. **OMMT.Plan** does not aim to cover fully the journey planning step. It provides services that can be useful to some players in specific multi-modal contexts:

- The UIC **MERITS** service is used by rail operators for the exchange of timetable data. It supports the GTFS format.
- The UIC **DRTF** fare distribution system is used by rail operators for the exchange of fare data.
- The UIC Real Time Mobility Data Exchange (**RTMDE**) is used by rail operators to exchange actual train status data.
- The OSDM **location API** *getPlaces,* when queried with the name of a *Place* or with its geocoordinates, returns one or more valid location codes and *Place* names (please refer to the following section for a description of the *Place* data model). A Service Provider can expose this API so that the API consumer may request whether the Provider serves a certain town or address and which location code should be used subsequently to request offers through **OMMT.Book**. TSGA offers a service, built according to OSDM *getPlaces* specifications, supporting Europe-wide locations relevant to multiple transport modes.

6. OMMT.Book Overview

The objective of the **OMMT.Book** API is the enablement of the online integration and distribution of multimodal mobility services. It is an open-source API based on UIC OSDM (Open Sales and Distribution Model). The full specifications of OSDM are available at https://osdm.io/.

For the purpose of OMMT, only a subset of OSDM is needed. The main concepts of OSDM, relevant for OMMT, are introduced in the next section.

6.1. OSDM main concepts

OSDM JSON-based APIs enable the interaction of transportation service providers with retailers, as well as with external providers of transportation services. The OSDM model defines the roles of different actors. For the purpose of the present Technical Guidelines, we will refer simply to the roles of API Provider and API Consumer.

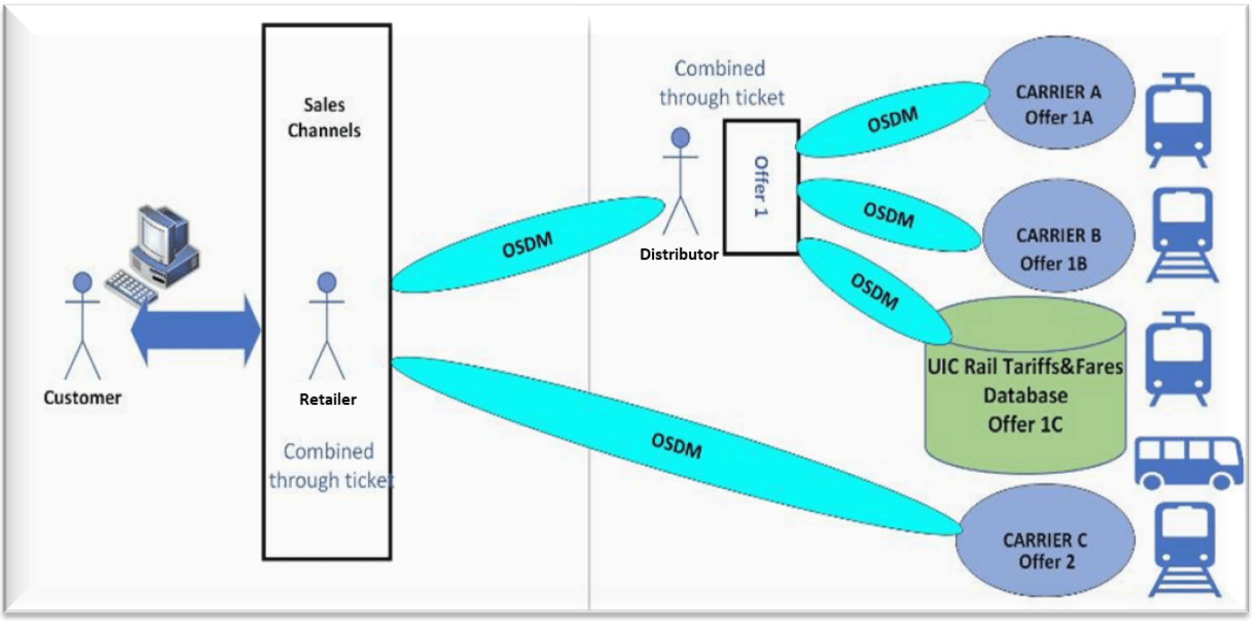

Figure 9 - Interactions enabled by OSDM

6.2. OSDM basic booking process flow

For the purpose of OMMT Version 1.0, the following basic OSDM shop-and-order process flow is of interest:

1. The API provider creates *Offers* for a *Trip* requested by the API consumer
2. The API provider provides a *Booking* based on a previously provided *Offer* selected by the API consumer
3. The API provider provides a *Fulfilment* (for example, a digital ticket) for a previously provided *Booking* confirmed by the API consumer.

Three functions of the OSDM API implement the above basic process steps:

1. *createOffers*
2. *postBookings*
3. *postFulfillments*

Offer requests made to *createOffers* do not necessarily contain the indication of a trip. Some products, particularly in a multimodal context, can be sold more efficiently with a non-trip-based approach:

- Multi-ride products valid for multiple trips, such as daily, weekly or monthly passes
- Single-trip tickets covering any trip within a specific area (if the customer is aware beforehand of the needed ticket, it is easier and faster to select it from a catalog, without indicating the trip)
- Reduction cards

In such cases the API consumer can request offers for products not associated to specific trip.

The above are the functions strictly required for the purpose of OMMT Version 1.0. The full OSDM API offers many more functions, including those for creating Trips, creating on-hold Offers, modify and delete Bookings and Fulfilments, manage Refunds and Complaints, get Coach Layouts, manage Reimbursements, and others. For OMMT Version 1.0 they represent optional functions that can be used depending on specific requirements.

6.3. OSDM essential data models

For the purpose of OMMT Version 1.0, the essential OSDM data models to be considered are the following:

- ✓ *Places*
- ✓ *Trips*
- ✓ *Offers*
- ✓ *Products*
- ✓ *Passengers*
- ✓ *Bookings*
- ✓ *Fulfillments*

OSDM specifications provide a representation of the data models underlying the API specifications. The full details are self-represented in the OpenAPI specifications. As such, some of the details of how the information is structured in the API are not represented or simplified in the data models. The main purpose of these data models is therefore to help a quicker understanding of the API and its underlying concepts.

Places

Places are resources representing a specific location in a *Trip*: departure, origin, intermediate stop or other. They can be of different types:

- *Address*: any street address can be represented here
- *PointOfInterest*: used to represent a specific point of interest
- *StopPlace*: represents a place where a train or a bus stops, through its code (*stopPaceRef)* and name (*stopPlaceName*). The code can come from different code sets. Using the UIC code set is highly recommended and mandatory for train stations. For example, as the UIC code for Zürich Hauptbahnhof is 8503000, OSDM represents it as `urn:uic:stn:8503000.`
- *GeoPosition*: allows indicating any location on the globe using its geographical coordinates.
- *FareConnectionPoint*: allows to model virtual border points by defining stations within the connection point lies.

Places are modelled in the API as resources with a long time-to-live, which should allow efficient caching of this data, therefore removing the need of getting full location details in transactional operations

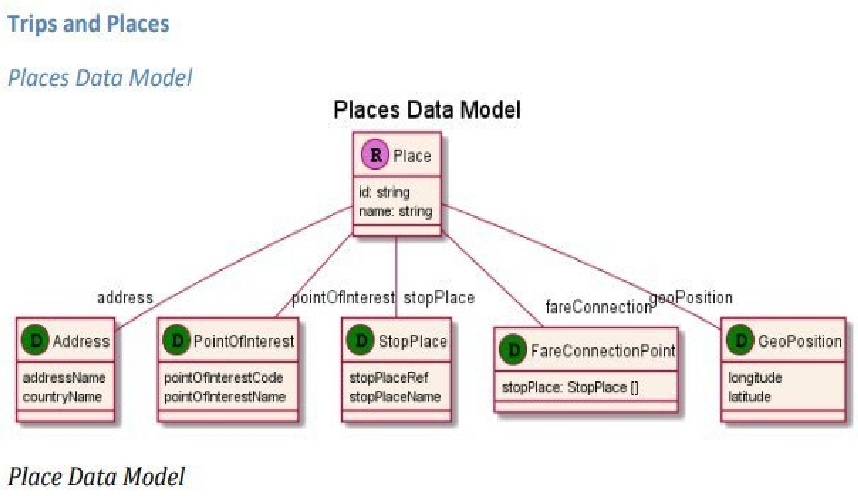

Figure 10 - Place data model

Trips

Trips represent the concrete realization of a trip. The Trip data model is aligned with the CEN Transmodel standard data model and the CEN Open API standard for distributed journey planning (OJP).

A *Trip* is composed of one or more *tripLegs* and can be of one the following types:

- *TimedLeg*: A type of leg with a timetable schedule, such as a leg provided by public transport
- *TransferLeg:* A type of leg that links two legs, such as walking from one stop to another
- *ContinuousLeg:* A type of leg that is not bound to a timetable, for example provided by an on-demand service (e.g., taxi, vehicle-sharing, etc)

Each *tripLeg* represents a connection between two *Places* where the traveller will either step in a transport vehicle or step out of a transport vehicle.

The departure *Place* of a Trip is the *origin,* while the arrival *Place* is the *destination*. A *Trip* can be in states *planned, confirmed, changed, or cancelled*.

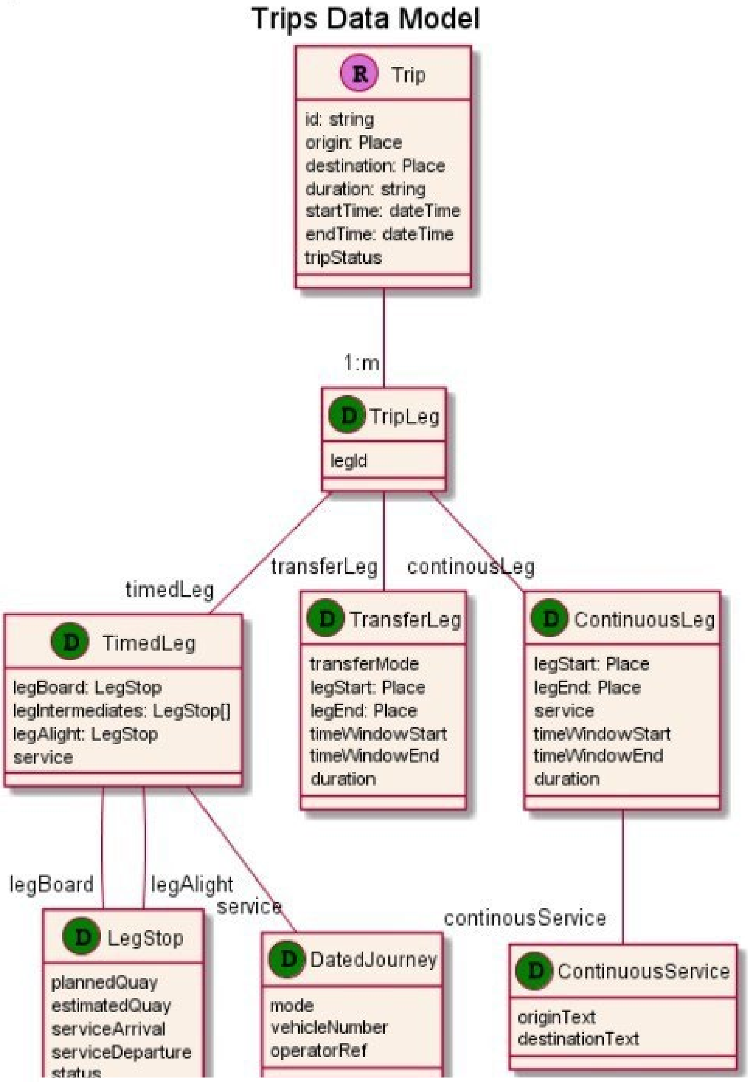

Figure 11 - Trips data model

Offers

The essential components of the Offers data model required for OMMT are described in the following.

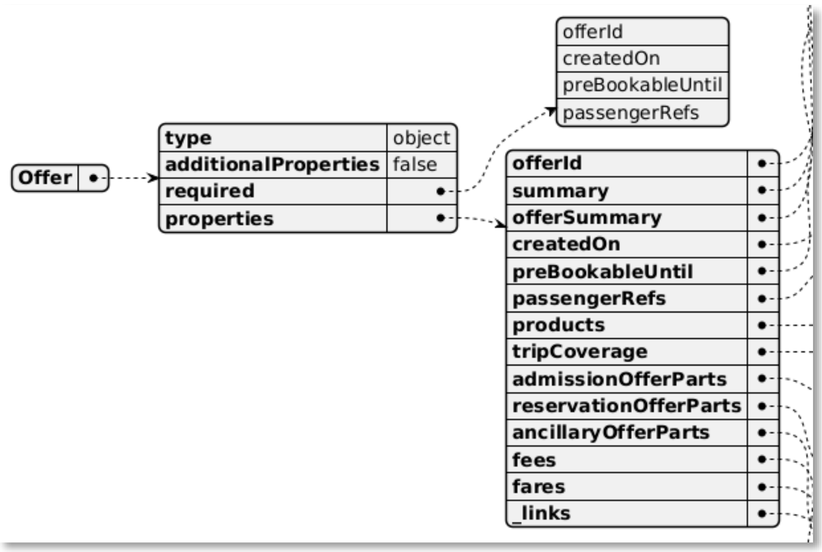

Figure 12 - Offer resource attributes

The API Provider may propose several *Offers* of different comfort and flexibility levels.

An *Offer* includes one or more *OfferParts. OfferParts* can be of different type, depending on what they represent:

- *Admissions*
- *Reservations*
- *Ancillaries*

However, all these different types share a significant amount of characteristics: they all apply to a defined set of *Passengers*, have a *Price* (calculated individually or collectively), and a few additional attributes.

Admissions represent a travel right, or the entitlement to travel onboard a vehicle (e.g., train, bus, ferry, etc) between the given origin and destination, following the given route, without a seat reservation. In some vehicles, seat reservations or an ancillary product (such as a WiFi connection or a meal onboard) can be associated with the admission for one or more of the *tripLegs*. A link will in this case point from the admission to the *Reservations* or *Ancillaries*, and the link will be qualified. *Ancillaries* can be either included or optional, while *Reservation* can also be mandatory to travel.

Reservations represent seat or other accommodation type reserved on the transportation. They might contain multiple seats/places. In contrast with admissions, a reservation is in essence bound to a specific train (or bus, etc), although it does not include the entitlement to board the train (or bus, etc). Passengers therefore typically need an associated admission offer part or other entitlement (such as a pass) in order to actually travel. *Reservations* have several additional attributes, such as information on the accommodation type and, once the offer will have been booked, the exact reserved places, with their properties and links to the concerned *Passengers*.

Ancillaries are used to represent non-transport products associated with the transportation offer. They could be onboard services such as a WiFi connection or a meal, or services associated with one of the stops, or origin/destination, like a parking spot or lounge access. This offer part is significantly simpler than those instantiating transport products, and has only one additional attribute, the category of the ancillary.

Moreover, *Offer* properties include:

- References to covered *Passengers* and *TripLegs*
- *Products* being offered

Products

Products are the products actually offered by the API Provider. *Products* resources contain all the conditions and attributes of the product, regardless of the actual sale case. Typically, this matches commercial products having the same name and recognizable common sales and after sales characteristics.

The *searchProducts* OSDM API allows the API consumer to search for products. Multiple parameters are supported, e.g., begin of validity, regions / zones where the offers are valid, flexibility. Product tags can be used in the search, offering high flexibility, including free-text requests, such as “I want to travel for two weeks in Germany” (if the provider supports AI search of its products).

Passengers

Passengers resources represent the passengers for whom the offers are proposed. All offers generated are always proposed for the complete set of passengers (no partial offers covering only a part of the passengers are generated). However, it is possible that because of age, reductions or other, some passengers are allowed to travel some *tripLegs* without actually needing a travel right or reservation. It is for example usually the case for infants traveling on their parents’ lap.

While several attributes can be set for passengers, only a few may be required at offer stage. The key elements at offer stage are already specified in the *Offer* request. The link between the possibly anonymous passenger profiles (in most basic form: a unique (in the booking) reference, age and reduction entitlement if any) created in the offer request and the passenger resources in the *Offers* can be made based on the passenger reference attribute. The *Passengers* resources created in the context of offers have their lifetime strictly limited to the lifetime of the offer resource they are part of. The resource and all local traces of it should be discarded once the offer has been booked or reached the end of the lifetime, in order to avoid any privacy concern.

The following elements are used to define a passenger profile:

- the passenger's age
- the reduction cards the passenger owns
- whether the passenger is a reduced-mobility or otherwise disabled passenger
- other specific status entitling to specific fares (military, senator, journalist…).

Booking

The *Booking* represents the *Offers* that have been selected and turned into a booking on request to the provider of the offers. It contains a set of sub resources, most of which were encountered in the offer stage, but also adds a few specific attributes and information, the most important being the booking *Status*. The booking will indeed evolve over time based on API consumer actions, time elapsed or other business events.

The *Booking* also contains additional attributes that are needed to manage and control the confirmation of the booking when it is in provisional state, such as the ticket time limit or the fulfillment options. The ticket time limit is the time during which the booking is guaranteed to remain available for confirmation for the price and possible reservations assigned at provisional booking time. Basically, it is the time given to the API consumer to perform all updates needed to confirm the booking and trigger that confirmation.

Also located at the root of the *Booking* structure is the ticket time limit. This is the time for which the provider will hold a booking in pre-booked state, waiting for the confirmation while guaranteeing the booking for the given products, spaces at the announced price. Obviously, this value only has a meaning for a booking in pre-booked state. A commonly accepted value would be around 30 minutes, which is normally sufficient to allow finalizing the booking, while not monopolizing resources too long in case the booking is abandoned without properly cancelling it. However, some systems may decide a longer time. Obviously, the value for the booking ticket-time limit can never exceed the earliest ticket time limit of any of its offer parts.

FulfillmentOptions allows the API consumer to specify the format desired for the fulfillment. Digital fulfillment only is considered in the OMMT scope.

BookedOffers are actually the same resources as the *Offers* except that they are now booked. Most of the resource remains unchanged, except for the sections on reservation details, where the sections related to the reserved places will now be populated with the references to the space allocated by the provider system where the transport product is hosted.

Fulfillments

Fulfillments could once have been called tickets. But the evolutions in the industry have led this to be a limitative naming, as various kinds of ticketless onboard controls are rapidly taking over and become the norm rather than the exception. From a distribution standpoint (the focus of OSDM), the only need is the possibility to point at a fulfillment representing an *OfferPart* for after sales operations. In most case it is a PDF document and/or a barcode.

6.4. Available resources

The full specifications of OSDM are available at https://osdm.io/, including:

- Specification document: *IRS 90918-10: Open Sales and Distribution Model OSDM*
- Open API specification in yaml format
- JSON schemas
- Test scenarios
- Sandboxes.

7. OMMT.Move Overview

As far as fulfilments are concerned, OMMT supports two models: a centralised model and a decentralised model.

- **Centralised:** The Orchestrator generates barcodes based on UIC FCB (Flexible Content Barcode) specifications and provides the Service Providers that need to validate or inspect the barcodes with tools (API, SDK or white label app) for reading the barcodes at validation/ inspection touchpoints.
- **Decentralised:** The Service Provider provides at sales time its own fulfilment (typically a barcode, but it could be an alphanumeric code as well), compatible with its own existing validation / inspection equipment and systems**.** The Orchestrator ensures that the Passenger has available the fulfillment when required during the trip. For example, the Orchestrator may provide a smartphone app that displays multiple barcodes for the same trip / account, depending on the location or on the Service Provider.

The centralised model is enabled by **OMMT.Move**, that is based on UIC FCB (Flexible Content Barcode) specifications. The detailed specifications are available in UIC IRS 90918-9. An implementation of encoding/decoding functions is available at:

https://github.com/UnionInternationalCheminsdeFer/UIC-barcode

The main contents of the FCB specifications are summarised in the following sections.

7.1. Conversion of data elements into barcodes

The process of conversion of data elements into barcodes is made up by three key steps, each of which contributes to the transformation of ticket information into a simplified barcode format.

1. Collect the data elements defining the ticket (fulfilment).
2. Convert the data collection into the FCB data structures and create the signature.

3a. Encode the binary chain into the Aztec binary content ("Aztec encoding").

3b. Convert this binary string into a graphical representation of the barcode (including error correction data).

7.2. Data composition

In the barcode, data are gathered in three logical groups:

- **a header:** the header makes it possible for a reader to determine which encoding is used and, thus, how the barcode data should be interpreted, and which (public) keys should be used to check the signature.

- **the ticket data:** the ticket data is the data that has to be converted into a barcode. The FCB specs specify how the data is to be coded or represented (also called "open data").
- **a seal/signature:** the seal is a digital signature preventing generating or modifying barcodes by others than recognised companies.

7.3. Signing and verifying the data

Data signing plays a pivotal role in ensuring the integrity, authenticity, and confidentiality of information. The process is divided into 5 distinct steps, which illustrate the methodology, tools, and key considerations involved

1. The first step is to generate the two cryptographic keys (private/public). This process takes place at least once every 18 months in the security provider's distribution system.
2. The private key is a cryptographic key that is uniquely associated with a public key and is not made public. The private key is used to compute a digital signature that can only be verified by the corresponding public key.
3. The private key is only known and generated by the computer that generates the key pair (private - public key). No (railway) staff should have access to the private keys. The tool that generates signatures must be able to use the private key while keeping it secret. The usual way to achieve this is by using a hardware security module (HSM).
4. The public key is a cryptographic key that is uniquely associated with a private key. It may be made public. The public key may be known by anyone and may be used to verify a digital signature that has been signed by the corresponding private key.
5. The public key must be published on the UIC PKMW website https://railpublickey.uic.org/. This does not have a negative impact on security.

The following figure illustrates the key steps involved in the secure authentication and validation of critical data.

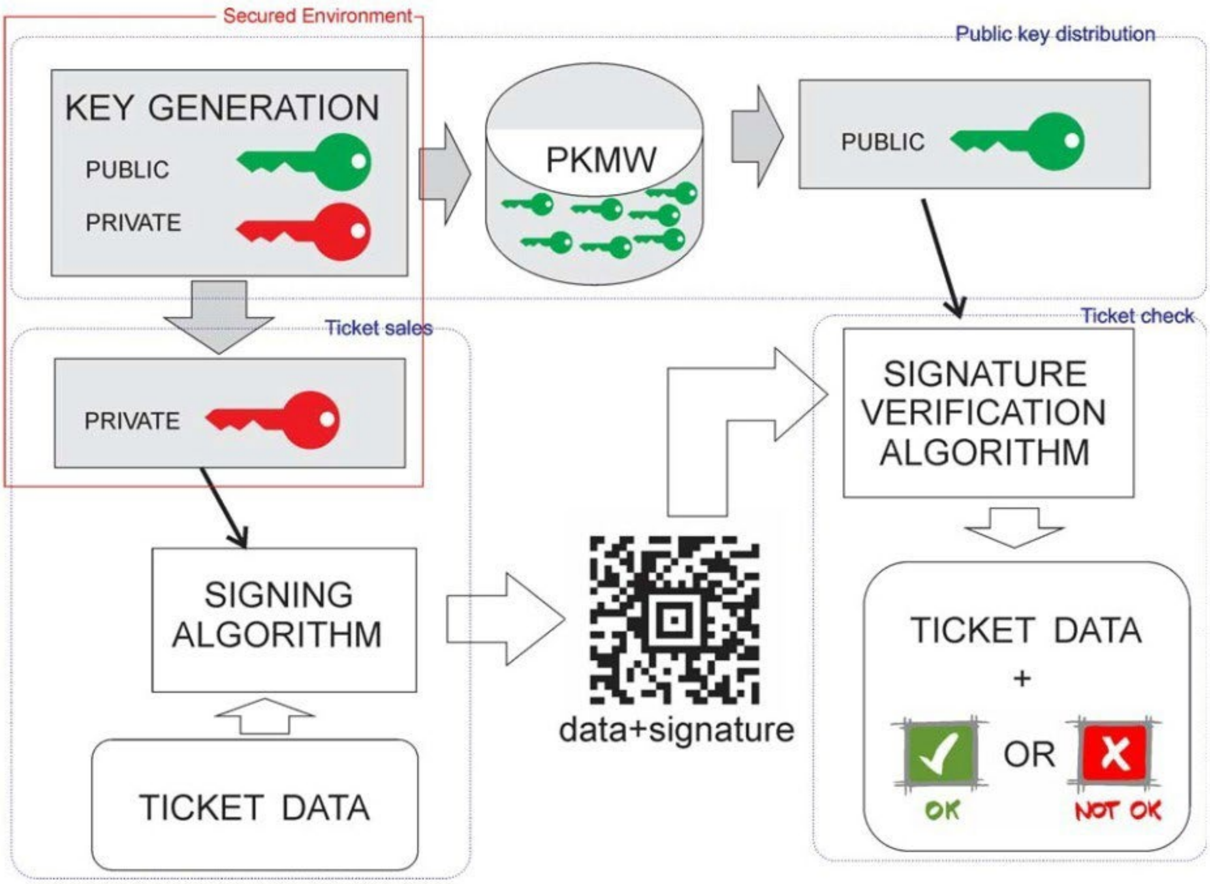

Figure 13 - Signing and verification with private and public key pair

7.4. Flexible Content Barcode (FCB)

A Flexible Content Barcode holds in encoded format (so that it can be interpreted by a machine) different kinds of data for different kinds of contracts and can be used in a number of contexts.

The data in the barcode is structured and, since the content is variable (depending on the kind of contract, for example), the barcode size is variable too.

The barcode itself is created in accordance with the Aztec standard.

Use cases – what can an FCB be used for

FCBs are more than just barcodes: they are enablers for secure and seamless transportation systems.

FCB support both **Security in Data (SiD)**, where authenticity is guaranteed by extra data present on the ticket, and **Security in System (SiS)**, where authenticity is guaranteed by data present in a remote system that can be accessed by means of a reference present on the ticket.
Possible use cases of Flexible Content Barcodes are listed below:

- Offline control of tickets without reference data on a device (SiD)

- Offline ticket inspection - reference data on the device (SiD)
- Online control (SiS)
- Partially-automated ticket checking (SiD)
- Opening platform gates (SiD, SiS)
- Opening station gates (SiD)
- Information function for routing in a station (SiS)
- Barcode to identify tickets for refunds (SiS)
- Barcode to identify tickets for after-sales processes (SiS)
- Validation of settlements vs control data (SiS)
- Annotation, "digital stamp" (SiD)

Ticket Data elements of the FCB

The ticket data elements embedded in a Flexible Content Barcode (FCB) are the following:

- *issuingDetail:* Data specific to the issuer and issue of the ticket (company, timestamp, place where the ticket is sold, specimen flag, payment details, etc.).
- *travelerDetail:* Details on the type of traveler(s) and their preferred language e.g. to display ticket information on the screen of an unmanned device in the event that a ticket is displayed on the screen of a check-in device or ticket vending machine, for example.
- *transportDocument:* The actual tickets/contracts. One or more of the following documents may be included: openTicket, pass, reservation, carCarriageReservation, voucher, customerCard, group ticket counterMark, parkingGround, fipTicket, stationPassage.
- *controlDetail:* The procedure for checking the ticket: whether passport identification is required, the percentage of tickets to be checked online at random (additional SiS), whether age verification is required (if the ticket is valid only for a specific age group), etc. Information text to be used for annotation purposes.
- *Extension:* Specific issuer data - defined bilaterally.

Encoding FCB Data

The absolute minimum set of records required to define a FCB 2D barcode is a header and a U_FLEX record type. The set of records is reduced in size by means of a compression technique and is then printed/displayed on the support medium (paper ticket, home-printed ticket, smartphone screen, etc.).

Number of sequence	Element	Number of characters	Mandatory	Code	Comments
1	Record ID	6A	X	X	Composition: • “**Uxxxxx**" for record types, **standard in UIC**."xxxxx" is variable and determines the type. • **RICS code** of four characters + "xx" for **record types defined by separate TCOs**. "xx" is variable and can be completed as desired by the TCO for each type (two characters).
2	Record version	2N	X		Allows **different versions of one record type** (with the same record ID).
3	Record length	4N	X		**Number of bytes** in the record data, starting from the first character of the "Record ID" element.
4	Record data	Data bytes			The **actual record data**

Number of sequence	Element	Number of characters	Mandatory	Code	Comments
1	Record ID	6A	X	X	**ID is "U_FLEX"** (UIC flexible, structured data barcode message).

2	Record version	2N	X		**Enables different versions to co-exist** (important in the case of pre-sales). Only the major version number is used in this numbering, the minor number is not used, nor the (possible) patch number. (e.g. 11.3.1: 11 = major number; 3 = minor number; 1 = patch number). Records with the same major number can be read by any device, able to read this major number. In case the reading is no longer possible with the old version, a new major number should be chosen.
3	Record length	4N	X		**Number of bytes** in the record data, starting from the first character of the "Record ID" element.
4	ASN.1 data	NBytes	X		**The actual data** as outlined in the chapter on data definition, bearing in mind the correct use case. The data in ASN.1 format follows the rule for canonical unaligned PER encoding.

7.5. Dynamic Flexible Content Barcode - DOSIPAS

SiD tickets have some security limitations. When a ticket is copied, it can also be used as a valid ticket.

To avoid this kind of misuse, tickets are often personalized. This personalization makes ticket check more complicated: not only the ticket has to be checked, but also an official document (like an id-card or a frequent traveller card) of the corresponding passenger.

A good alternative to add extra security features to avoid copying is to show the barcode on a smartphone and to add a dynamic part in the data that is translated in the barcode. This results in a so-called "dynamic barcode".

The whole process is as follows:

1. App on the client device generates a keypair ("dynamic keypair")

2. The dynamic public key is added to the ticket data, to generate a static signature (this calculation is done on the server of the security provider as before)

3. Dynamic data is generated

4. All the previous data is signed, using the dynamic private key

5. Header + ticket data + dynamic public key + static signature + dynamic data + dynamic signature are translated into an AZTEC 2D barcode

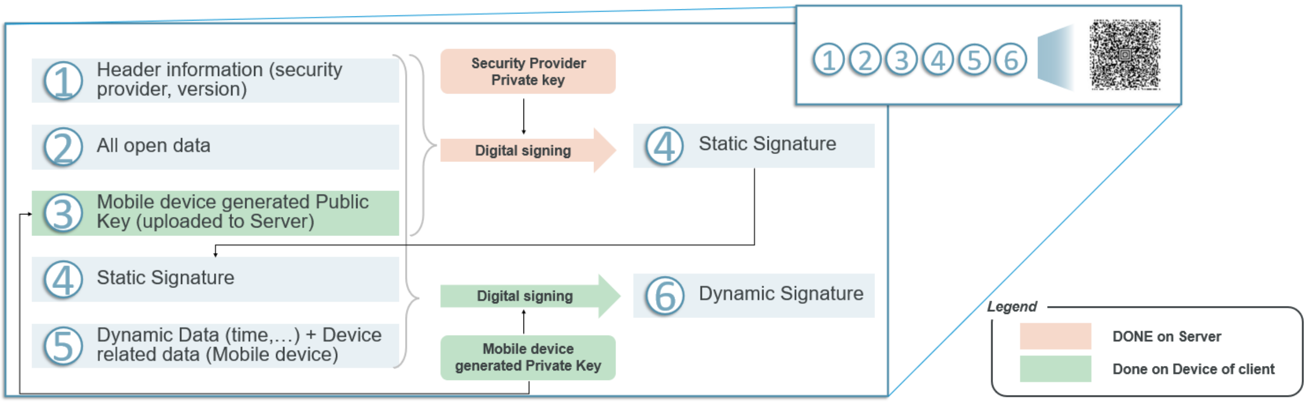

Figure 14 - Generating a Dynamic Flexible Content Barcode

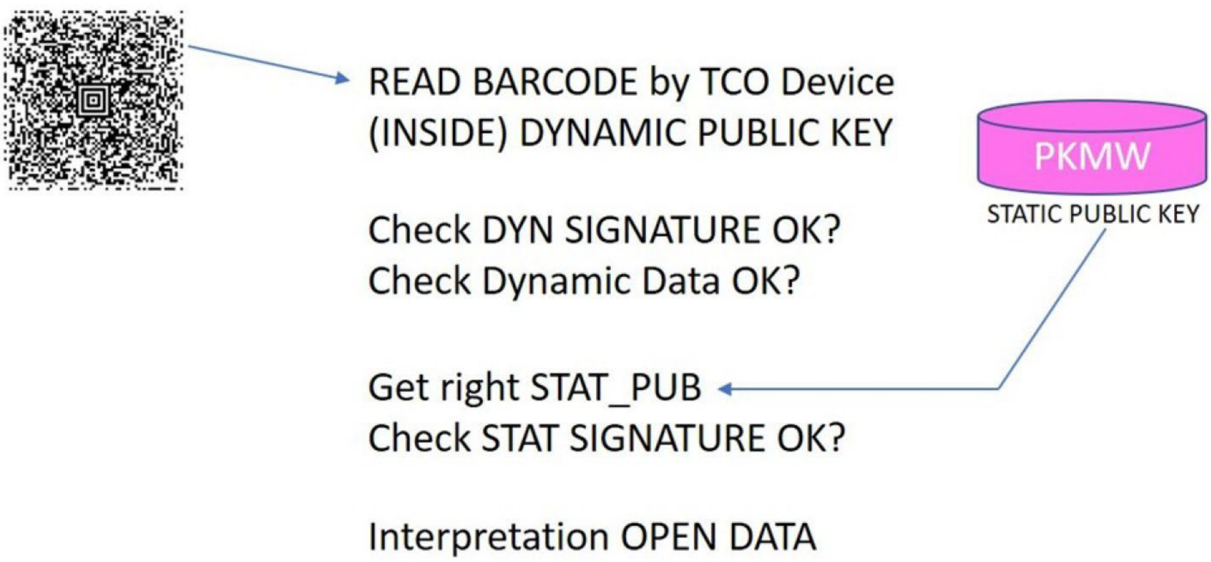

Figure 15 - Verifying a Dynamic Flexible Content Barcode

Terms and Definitions:

- **Static Private Key**: Key, used to generate the static signature.

- **Static Public Key:** Key, used to check the static signature, downloaded by the TCO from the PKMW.
- **Static key pair:** Keys (static private key + static public key) are generated on the central servers of the security provider.
- **Static Signature:** Signature, calculated with a digital signature algorithm, generated on the central servers of the security provider.
- **Dynamic Private Key:** Key, used to generate the dynamic signature.
- **Dynamic Public Key:** Key, used to check the dynamic signature. This key is included in the barcode itself (is part of the header of the barcode).
- **Dynamic key pair:** Keys (dynamic private key and dynamic public key) are generated on the app on the mobile phone of the client (app developed by the ticket seller, allowing the activation of the ticket and performing the "dynamization" of the ticket).
- **Dynamic Signature:** The dynamic signature itself, generated with a digital signature algorithm on the mobile device of the client.

While FCB has a standard header, the version 2 of FCB (FCBv2) rather than a header has a "Double Signed Package Structure" (**DOSIPAS**).

It is called "double signed package" since there are 2 signatures to protect the data (the "package"). One signature is calculated on static info (not changing during successive checks) and made on the central servers of the security provider. The other signature is calculated on info that can change during successive checks, reason why this one is called the dynamic signature.

As the static data and the corresponding signature is encapsulated in the dynamic signature, which will only be dynamic if the data is effectively changing, it makes sense to talk about Level 1 and Level 2 signatures.

7.6. Key management

As the DSA signature algorithm is an asymmetric and robust cryptographic procedure, the key pairs generally need to be changed only rarely. A relatively long validity period can be applied. Public keys can be published on the **UIC PKMW (Public Key Management Website).** All of the details of the processes for distribution of security providers' public keys to ticket inspectors are outlined in the UIC document "Public Key Distribution using PKMW". The issuing company should generate a private-public key at least every 18 months.

The security provider should provide details of its public key on the website at least three months before production so that the TCO can download them in advance. The "startDate" is the first possible day of travel. The "endDate" is the end date for signature validity, the last possible day of travel. The maximum duration of validity of the keys (e.g. 12 months) is defined in each bilateral agreement. Tickets may be valid over longer periods (e.g. two months or more) and only one signature per ticket is possible, therefore the validity periods of subsequent keys must overlap. The overlap period must be at least as long as the maximum ticket validity time span (minus one day).

The following graph showcase an example showing minimum key validity overlap.

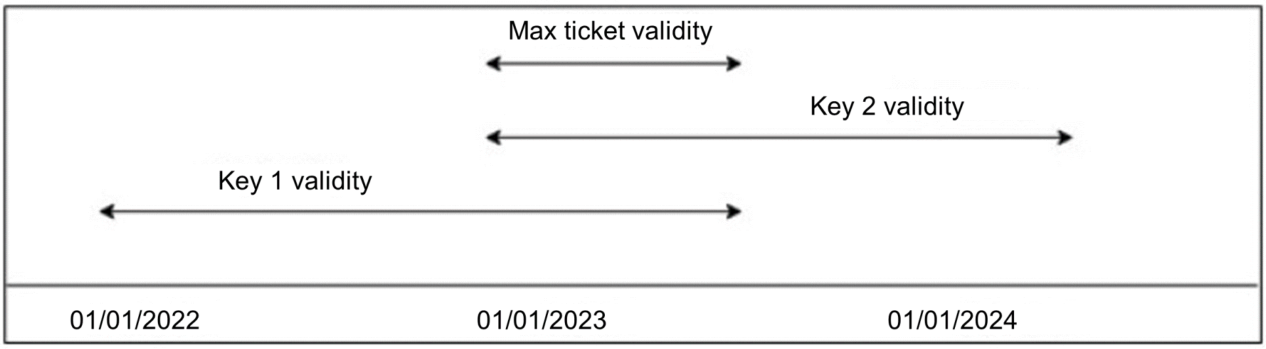

Figure 16 - Keypair validity example

7.7. Available resources

UIC has made available detailed specifications in UIC IRS 90918-9.

An open implementation of encoding/decoding functions, that can be integrated in mobility operators' apps, is available at: https://github.com/UnionInternationalCheminsdeFer/UIC-barcode.

8. OMMT.Sync Overview

OMMT.Sync is based on the specifications provided by UIC IRS 90918-4 "e-Ticket Exchange for Control". These specifications deal with the capability of controlling dematerialised tickets (e-tickets) that are not necessarily linked to a reservation (non-integrated reservation tickets).

OMMT supports two models for Shared Data Services:

- **Centralised Model** - Services are provided by UIC eTCD (electronic Ticket Control Database), an available cloud-based centralised common service built according to the above specifications.
- **Decentralised Model** – Services are provided by the Orchestrator according to the above specifications.

The Centralised Model is recommended, as it is already up and running (it is operational since February 2020 and is used by an increasing number of RUs to control international tickets), it maximises synergies and enables sharing among multiple companies.

8.1. eTCD process and functions

The following diagram illustrates the actors and their interactions.

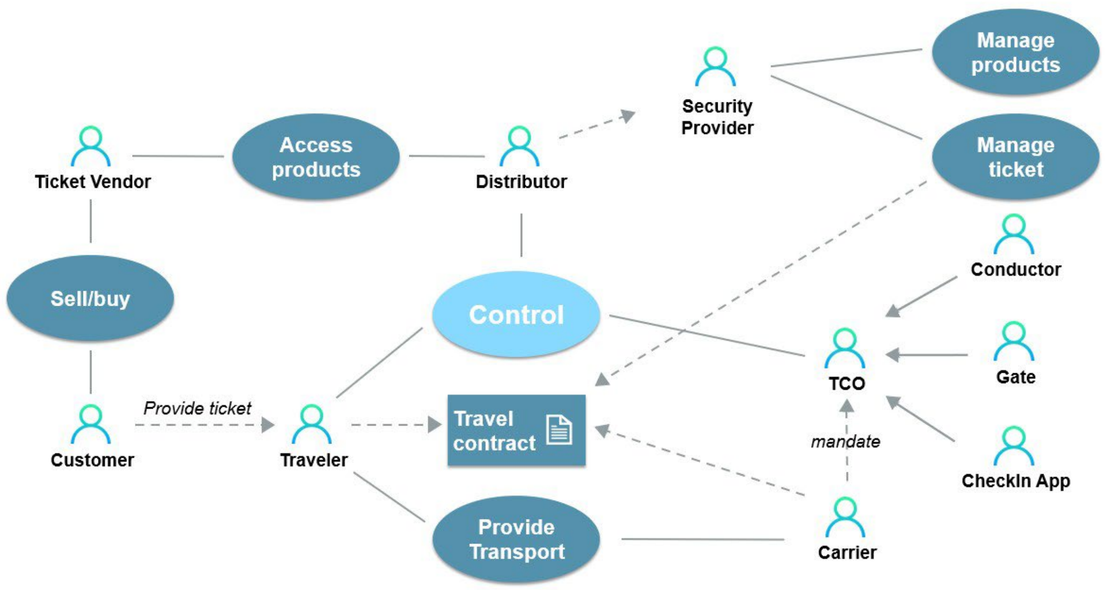

Figure 17 - Actors and Interactions in eTCD

The following business capabilities supported by eTCD are essential for OMMT:

1. **Online Control**: The e-ticket needs to be validated by the Ticket Controlling Organisation (TCO). In order to achieve this, all data describing the ticket and all previous annotations made to it must be provided to the TCO. The data needed to identify the ticket ("ticket identification") must be transferred to the TCO. The functional use cases that provide this capability are:
   - Retrieve a ticket (or pass)
   - Provide tickets
   - Provide control annotation

2. **Travel contract modification**: The TCO can make changes to the ticket to change the validity concerning the time, the route, the allowed products and the number of passengers. The functional use cases that provide this capability are:
   - Provide validity change annotation
   - Provide quality violation annotation (delays, service degradation)

3. **Check-in and check-out**: The functional use cases defined to support this capability are:

- Provide check-in annotation
- Provide check-out annotation

eTCD supports many more business capabilities, that for OMMT Version 1.0 represent optional functions that can be used depending on specific requirements, including:

- Retrieve tickets of a train
- Validity Extension
- Comfort check-in
- Mandatory check-in on full flex ticket
- Inspection mark from gates
- Delay notification
- Declaration of non-use of ticket
- Class upgrade or downgrade
- Send messages to distributor / issuer / train manager
- Night train use cases

eTCD allows the sharing of information about the lifecycle of the ticket among all interested parties in almost real-time.

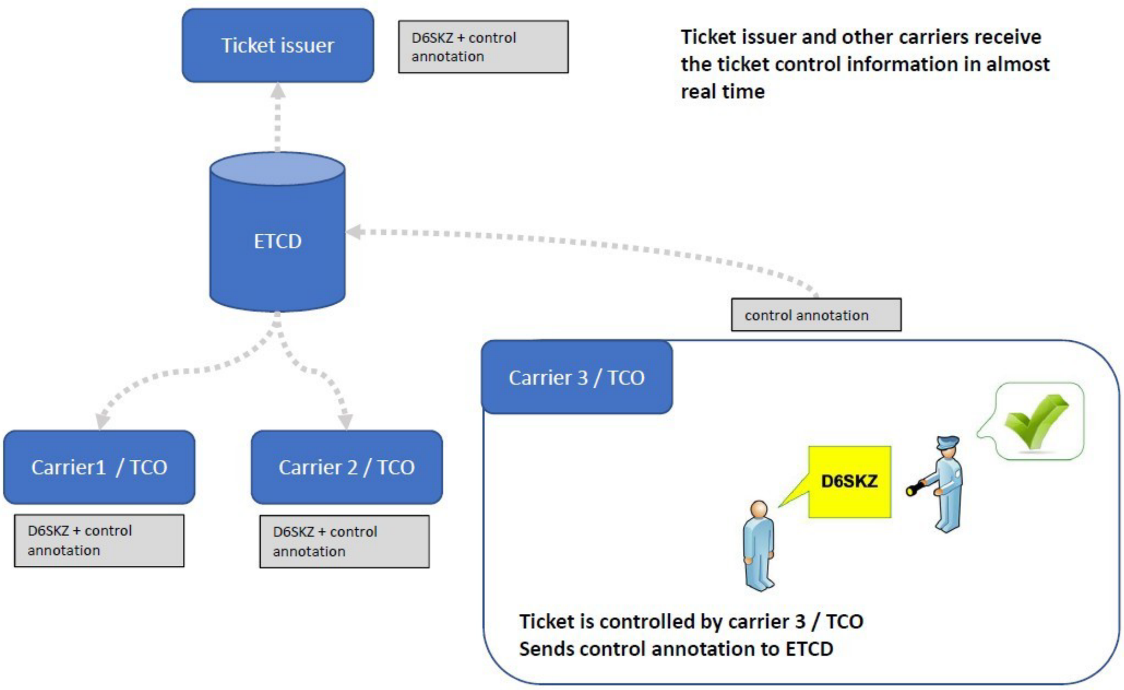

Figure 18 - Almost real-time sharing of annotations enabled by eTCD

For the purpose of OMMT, the essential functions provided by the eTCD API are:

- *postTicket:* Creates a new e-ticket.
- *getTicket:* Returns a e-ticket, including annotations, given a ticketId.
- *createAnnotation:* Creates an annotation to a ticket.

8.2. Available resources

UIC can provide:

- Specifications (published as *IRS 90918-4 e-Ticket Exchange for Control*)
- Implementation guide
- User's guide
- JSON schema
- Open API specification in yml format

9. OMMT.Share Overview

OMMT.Share is based on the specifications developed by UIC Next Generation Revenue Sharing (NGRS) project. NGRS specifications are available on request as they are not yet published. UIC is evaluating the release of an IRS with the NGRS specifications and, as a following step, the implementation of a common centralised service based on the specifications.

The NGRS specifications describe the process flow, data exchanges and APIs that enable the apportionment of revenues among partners in multimodal and MaaS contexts. The apportionments are the input for subsequent settlements (out of scope of NGRS), that may be either bilateral or enabled by a common clearinghouse. NGRS specifications cover also the generation of Revenue Accounting flows compliant with UIC IRS 30301 specifications.

NGRS specifications support several business models. The following are relevant for OMMT Intermodal Revenue Apportionment Services.

- **Pre-purchased ticket – commission**: Multimodal sales when apportionment is pre-defined by partners and can be computed at sales time, based on commission or agency fee and/or by pre-agreed share.

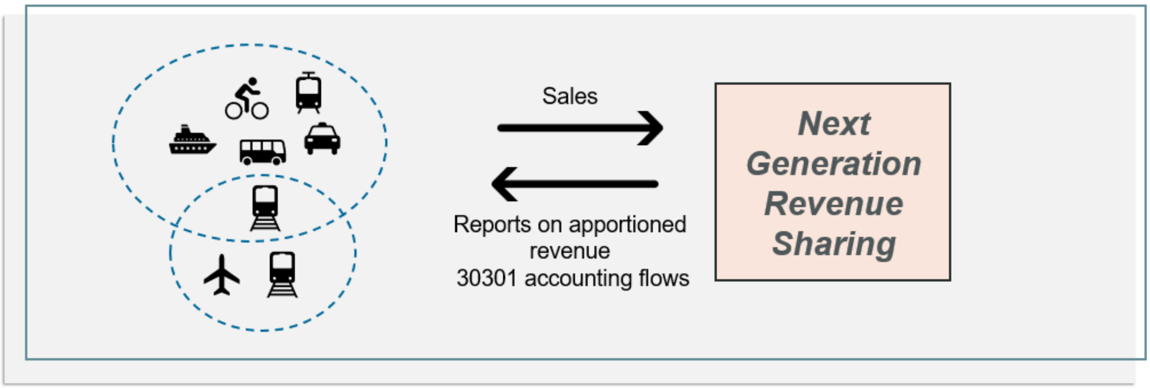

Figure 19 - NGRS model for Pre-purchased ticket – commission

- **Pre-purchased pass – pro-rata**: Multimodal pass or mobility package when apportionment is based on agreed pro-rata (data about usage are logged in order to periodically update the pro-rata, to be agreed by the partnership).

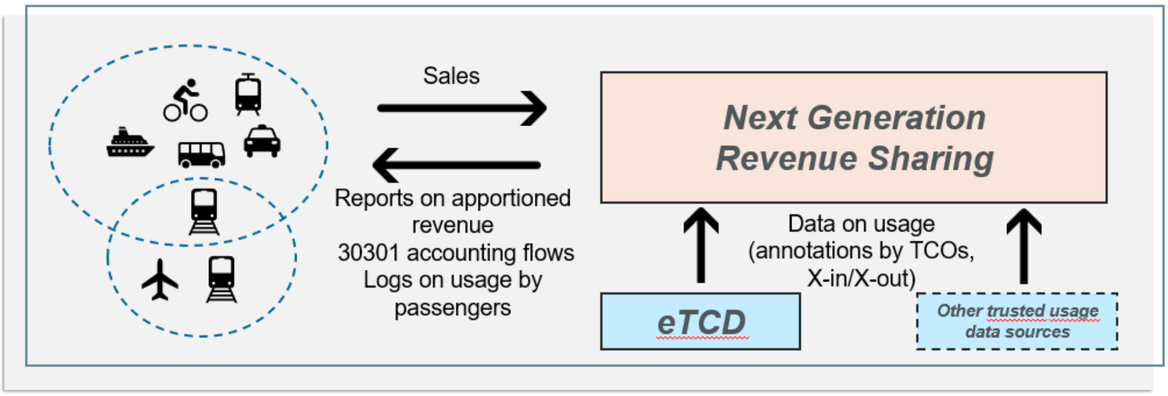

Figure 20 - NGRS model for Pre-purchased pass – pro-rata

- **Pre-purchased pass – usage**: Multimodal pass or mobility package when apportionment is based on actual usage of mobility services by pass holder (data about usage are logged and at the end of validity of the pass revenue is apportioned by an algorithm that analyses the logs; the algorithm is implemented by the partnership that makes it available via a API).

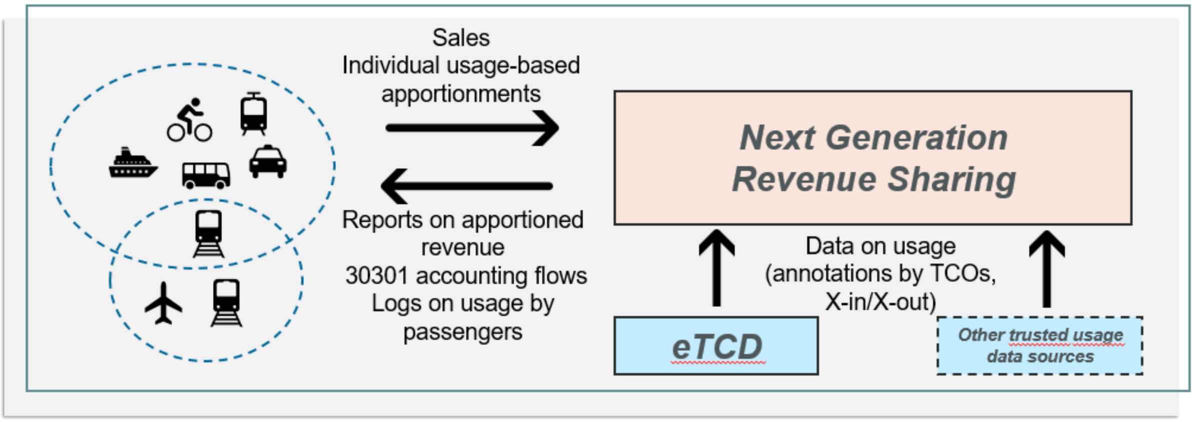

Figure 21 - NGRS model for Pre-purchased pass – usage

- **Account-based model**: Account-based multimodal partnership when apportionment is based on actual usage of mobility services by account holder (data about usage are logged and periodically the amount to be charged to the customer and the revenue is apportioned by an algorithm that analyses the logs; the apportionment API is to be provided by partnership; the partnership is responsible for charging the customer).

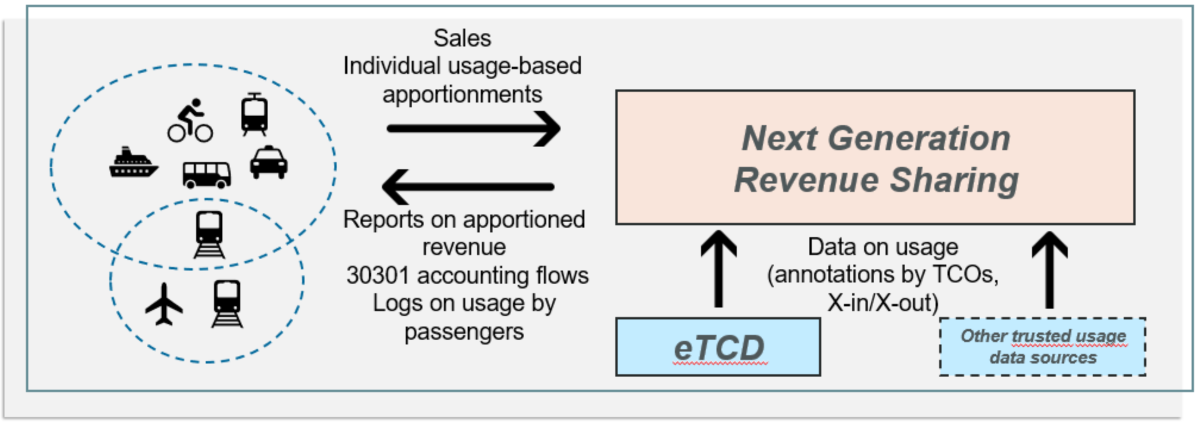

Figure 22 - NGRS model for Account-based model

9.1. Available resources

UIC can provide upon request the following NGRS documentation:

- Essential Guide to NGRS
- Introduction to NGRS
- NGRS Implementation Guide
- NGRS YAML specifications

10. Quick guides

This chapter explains how OMMT can be adopted by different actors. Specific focus is put on Minimum Viable Products (MVPs), i.e. on how to OMMT can enable basic multimodal and MaaS capabilities. Information on how to enable capabilities that go above MVPs is provided in more detail in previous chapters.

Each quick guide explains how OMMT enables **three main steps** required for delivering multimodal integrated mobility experiences to customers:

1. Booking
2. Journey
3. Revenue settlement.

The quick guides **do not address specifically the Journey Planning step**. Solutions for door-to-door journey planning are widely available. Standards (*de jure* or *de facto)* for data sharing required for journey planning are well known. The main standards for the different modes are mentioned in the following sections. The

Booking step enabled by OMMT can be fed with trip data (ie itinerary) generated by any journey planner. The *Trip* data model of **OMMT.Book**, based on OSDM, is consistent with the CEN Transmodel standard data model and the CEN OPEN API standard for distributed journey planning (OJP). **OMMT.Plan** supports the Journey Planning step by providing a function useful to get location codes.

The quick guides refer to **roles of Orchestrator and of Service Provider**. The quick guides for Service Providers depend on the mode of transport. It is assumed that a **commercial agreement is in place** between the Orchestrator and the Service Providers. As described in chapter 3.1, the same entity can fulfil both roles. For example, according to the first Multimodal Partnership Model described in chapter 3.1, one of the Service Providers, for example a Rail Undertaking**,** acts as the Orchestrator.

10.1. Quick guide for Orchestrators

The implementation activities required of the Orchestrator, collaborating with partner Providers, include:

- Integrate in its own sales and reservation application the API built according to **OMMT.Book** API services exposed by the partner Providers and consume, depending on the Use Case as detailed in the following sections, the *createOffers, postBookings and postFulfillments* functions.
- If the Orchestrator acts also as Retailer, enable the customer interface to handle multimodal use cases. If the Retailer is a different entity from the Orchestrator, expose to the Retailer sales APIs built according to **OMMT.Book.**
- Apply **OMMT.Move** specifications to generate FCB barcodes representing multimodal contents, depending on the Use Case.
- Expose to the partner Providers the following **OMMT.Sync** API functions, depending on the Use Case as detailed in the following sections: *getTicket, createAnnotation.*
- Ensure accounting of revenue apportionments according to commercial agreements and apply **OMMT.Share** to generate internal Revenue Accounting flows and to provide detailed reports on revenue apportionments to partners.

10.2. Quick guide for Scheduled Transport Operators (bus, metro, rail, ferries)

This quick guide section is addressed to public or private operators of scheduled transportation services that act as Service Providers.

The **Journey Planning step** can leverage available standards for exchanging static information (NetEx, the CEN standard adopted in Europe, or GTFS, a global industry standard) or real-time information (the CEN standard SIRI or the global industry standard GTFS-RT). **OMMT.Plan** can be used to provide location codes to the Orchestrator.

The adoption of OMMT depends on the scenario, as described in the following.

Scenario 1

In this scenario, the service provided by the Transport Operator **does not require a reservation** and the Integrated Mobility Experience Model is either **One Ticket** or **One Account**, where the integrated ticket / account is provided by the Orchestrator. The implementation effort required of the Transport Operator can be zero or minimal.

Step	What is required of the Operator
Booking	No implementation effort is required of the Operator. The Orchestrator issues the digital ticket / account and provides the fulfilment to the traveller.
Journey	To enable the validation / inspection of the traveller, the Operator has to integrate in its own validation / inspection application the **OMMT.Move** decoding function that allows to decode the FCB barcode (an Android implementation is available here https://github.com/UnionInternationalCheminsdeFer/UIC-barcode). For additional security, to enable the retrieval of the online digital ticket / account the Operator has to integrate in its own validation / inspection app the **OMMT.Sync** API provided by the Orchestrator and consume the *getTicket* API. If the commercial agreement requires it, to enable the sharing of validation / inspection data, the Operator has to integrate in its own validation / inspection app the **OMMT.Sync** API provided by the Orchestrator and consume the *addAnnotation* API. All the above implementation efforts of the Operator can be avoided if the Orchestrator provides the Operator with a (white label) app that includes the above functionalities, that the Operator can use in lieu of its own inspection app.
Revenue Settlement	No implementation effort is required of the Operator. The Operator will receive periodically a detailed report from **OMMT.Share**.

Scenario 2

In this scenario, the service provided by the Transport Operator **does require a reservation** and the Integrated Mobility Experience Model is either **One Ticket** or **One Account**, where the integrated ticket / account is provided by the Orchestrator.

Step	What is required of the Operator
Booking	To enable the reservation of its resources (e.g., a seat on a bus or ferry or train), the Operator has to expose to the Orchestrator the following functions of its own reservation application, as specified by the **OMMT.Book**: *createOffers* to return offers for a specific trip, and *postBookings* to create a booking based on a previously requested offer. The Orchestrator issues the digital ticket / account and provides the fulfilment to the traveller.
Journey	To enable the validation / inspection of the traveller, the Operator has to integrate in its own validation / inspection application the **OMMT.Move** decoding function that allows to decode the FCB barcode (an Android implementation is available here https://github.com/UnionInternationalCheminsdeFer/UIC-barcode). For additional security, to enable the retrieval of the online digital ticket / account the Operator has to integrate in its own validation / inspection app the **OMMT.Sync** API provided by the Orchestrator and consume the *getTicket* API. If the commercial agreement requires it, to enable the sharing of validation / inspection data, the Operator has to integrate in its own validation / inspection app the **OMMT.Sync** API provided by the Orchestrator and consume the *addAnnotation* API. All the above implementation efforts of the Operator can be avoided if the Orchestrator provides the Operator with a (white label) app that includes the above functionalities, that the Operator can use in lieu of its own inspection app.
Revenue Settlement	No implementation effort is required of the Operator. The Operator will receive periodically a detailed report from **OMMT.Share**.

Scenario 3

In this scenario, the Integrated Mobility Experience Model is either **One Purchase** or **One Media.** The digital ticket and the fulfilment are issued by the Operator.

Step	What is required of the Operator
Booking	To enable the booking of its resources (which may require or not a reservation), the Operator has to expose to the Orchestrator the following functions of its own sales application, as specified by the **OMMT.Book** API: *createOffers* to returns offers for a specific trip, *postBookings* to create a booking based on a previously requested offer, and *postFulfillments* to confirm the booking and trigger the fulfilment of the booking. This enables the Orchestrator to receive the fulfilment and provide it to the traveller.
Journey	To enable the validation / inspection of the traveller, no implementation effort is required of the Operator, that can use its own validation / inspection systems.
Revenue Settlement	No implementation effort is required of the Operator. The Operator will receive periodically a detailed report from **OMMT.Share**.

10.3. Quick guide for Bike / Scooter / Car Sharing Operators

This quick guide section is addressed to vehicle sharing operators that act as Service Providers.

The **Journey Planning step** can leverage available standards for exchanging vehicle sharing information: **NetEx**, the CEN standard adopted in Europe; for a bicycle and scooter sharing, **GBFS** (General Bikeshare Feed Specification), the open data standard for shared mobility, that supports both free-float and services where vehicles are docked at specific locations; **IXSI-5** is an API Specification developed in Germany for shared cars. In this quick guide it is assumed that the Journey Planning step allows the Orchestrator to identify trips including the detailed coordinates of the shared vehicles, that can be displayed on a map to the traveller.

Scenario 1

In this scenario, the Integrated Mobility Experience Model is either **One Ticket** or **One Account**, where the integrated ticket / account is provided by the Orchestrator. Reservation may be included or not; in the latter case, the user would directly unlock the bike / scooter when they need to use it.

Step	What is required of the Operator
Booking	The Operator has to expose to the Orchestrator the following functions of its own reservation application, as specified by **OMMT.Book**: *createOffers* to return offers for a specific trip, *postBookings* to create a booking based on a previously requested offer and *postFulfillments* to confirm the booking and trigger the fulfilment (the fulfilment may consist of an unlock code to be supplied to the traveller or may directly unlock the asset for immediate usage). The operator can also expose the *getAvailabilitiesContinuousServices* **OMMT.Book** API service that specifically supports on-demand mobility services, including vehicle sharing: it allows the Orchestrator to provide as input geo position and types of vehicle (e.g., bicycle, electric bicycle, cargo bicycle, scooter) and get in response a list of available pick-up places. The response can indicate the number and type of available vehicles and the pre-allocated time limit.
Journey	To communicate to the Orchestrator the check-in when the bike / scooter is unlocked and the check-out when the bike / scooter is released, the Operator has to integrate in its own application the **OMMT.Sync** API provided by the Orchestrator and consume the function *addAnnotation.* As an alternative, the Operator may expose the following **OMMT.Book** API services to be consumed by the Orchestrator: ▪ *getContinuousServiceUsageId,* that returns the on-demand service usage; ▪ *patchContinuousServiceUsageId,* that allows to start or end the usage of an on demand vehicle (this service supports also the sharing of a photo of the parking at the end of the service, if required).
Revenue Settlement	No implementation effort is required of the Operator. The Operator will receive periodically a detailed report from **OMMT.Share**.

<u>Scenario 2</u>

In this scenario, the Integrated Mobility Experience Model is either **One Purchase** or **One Hub**, where the ticket / account is provided by the Operator.

Step	What is required of the Operator
Booking	The Operator has to expose to the Orchestrator the following functions of its own reservation application, as specified by **OMMT.Book**: *createOffers* to return offers for a specific trip, *postBookings* to create a booking based on a previously requested offer and *postFulfillments* to confirm the booking and trigger the fulfilment (the fulfilment may consist of an unlock code to be supplied to the traveller or may directly unlock the asset for immediate usage). The operator can also expose the *getAvailabilitiesContinuousServices* **OMMT.Book** API service that specifically supports on-demand mobility services, including vehicle sharing: it allows the Orchestrator to provide as input geo position and types of vehicle (e.g., bicycle, electric bicycle, cargo bicycle, scooter) and get in response a list of available pick-up places. The response can indicate the number and type of available vehicles and the pre-allocated time limit.
Journey	In general, no implementation effort is required of the Operator. The service is charged to the traveller by the Operator. Should the commercial agreement include a fee due to the Orchestrator depending on usage, the Operator may expose the following **OMMT.Book** API services to be consumed by the Orchestrator: ▪ *getContinuousServiceUsageId,* that returns the on-demand service usage.
Revenue Settlement	No implementation effort is required of the Operator. The Operator will receive periodically a detailed report from **OMMT.Share**.

10.4. Quick guide for Taxi / Ride-hailing / Demand Responsive Transport Operators

This quick guide section is addressed to taxi and ride-hailing operators that act as Service Providers. It also applies to Public Transport Operators that provide Demand Responsive Transport services.

In this case, the **Journey Planning step** may not require interactions between the Orchestrator and the Operator, as journey planners may often provide a reasonable estimate of the duration of a trip leg by taxi or ridesharing. The General On-Demand Feed Specification, or **GOFS,** has developed a data standard aimed at demand-responsive transportation like on-demand transit, paratransit, taxis, and ride-hailing.

Scenario 1

In this scenario, the Integrated Mobility Experience Model is either **One Ticket** or **One Account**, where the integrated ticket / account is provided by the Orchestrator.

Step	What is required of the Operator
Booking	The Operator has to expose to the Orchestrator the following functions of its own reservation application, as specified by **OMMT.Book**: *createOffers* to return offers for a specific trip, *postBookings* to create a booking based on a previously requested offer and *postFulfillments* to confirm the booking and trigger the fulfilment (the fulfilment may consist of a car code to be supplied to the traveller). The operator can also expose the *getAvailabilitiesContinuousServices* **OMMT.Book** API service that specifically supports on-demand mobility services, including taxi and ride-hailing; it allows the Orchestrator to provide as input geo position and types of vehicle and get in response a list of available pick-up places with times of arrival.
Journey	To communicate to the Orchestrator that the trip leg has ended and the cost of the trip leg (in case the cost is not pre-paid and communicated at booking time), the Operator has to integrate in its own application the **OMMT.Sync** API exposed by the Orchestrator and consume the function *addAnnotation.* As an alternative, the Operator may expose the following **OMMT.Book** API services to be consumed by the Orchestrator: ▪ *getContinuousServiceUsageId,* that returns the on-demand service usage.
Revenue Settlement	No implementation effort is required of the Operator. The Operator will receive periodically a detailed report from **OMMT.Share**.

Scenario 2

In this scenario, the Integrated Mobility Experience Model is either **One Purchase** or **One Hub**, where the cost of the ride is charged by the Operator to the traveller.

Step	What is required of the Operator
Booking	The Operator has to expose to the Orchestrator the following functions of its own reservation application, as specified by **OMMT.Book**: *createOffers* to return offers for a specific trip, *postBookings* to create a booking based on a previously requested offer and *postFulfillments* to confirm the booking and trigger the fulfilment (the fulfilment may consist of a car code to be supplied to the traveller). The operator can also expose the *getAvailabilitiesContinuousServices* **OMMT.Book** API service that specifically supports on-demand mobility services, including taxi and ride-hailing; it allows the Orchestrator to provide as input geo position and types of vehicle and get in response a list of available pick-up places with times of arrival.
Journey	In general, no implementation effort is required of the Operator. The service is charged to the traveller by the Operator. Should the commercial agreement include a fee due to the Orchestrator depending on usage, the Operator may expose the following **OMMT.Book** API services to be consumed by the Orchestrator: ▪ *getContinuousServiceUsageId,* that returns the on-demand service usage.
Revenue Settlement	No implementation effort is required of the Operator. The Operator will receive periodically a detailed report from **OMMT.Share**.

10.5. Quick guide for Parking Operators

This quick guide section is addressed to parking operators that act as Service Providers.

The **Journey Planning step** would typically need static information about location and operators of parking facilities located nearby public transport stations. Some standards for sharing such information are available, like **APDS** from the Alliance for Parking Data Standards.

The adoption of OMMT depends on the scenario, as described in the following.

Scenario 1

In this scenario, the parking **is not reserved in advance** and the Integrated Experience Model is either **One Ticket** or **One Account**, where the integrated ticket / account is provided by the Orchestrator. The implementation effort required of the Parking Operator can be zero or minimal.

Step	What is required of the Operator
Booking	No implementation effort is required of the Operator. The Orchestrator issues the digital ticket / account and provides the fulfilment to the traveller.
Journey	To enable the validation / inspection of the traveller, the Operator has to integrate in its own validation / inspection application the **OMMT.Move** decoding function that allows to decode the FCB barcode (an Android implementation is available here https://github.com/UnionInternationalCheminsdeFer/UIC-barcode). If the parking service is not pre-paid, to communicate to the Orchestrator the check-in when the traveller's vehicle enters the parking and the check-out when vehicle exits, the Operator has to integrate in its own application the **OMMT.Sync** API exposed by the Orchestrator and consume the function *addAnnotation.* All the above implementation efforts of the Operator can be avoided if the Orchestrator provides the Operator with a (white label) app that includes the above functionalities, and the Operator can use in lieu of its own inspection app.
Revenue Settlement	No implementation effort is required of the Operator. The Operator will receive periodically a detailed report from **OMMT.Share**.

Scenario 2

In this scenario, the parking **is reserved in advance** and the Integrated Experience Model is either **One Ticket** or **One Account**, where the integrated ticket / account is provided by the Orchestrator.

Step	What is required of the Operator
Booking	The Operator has to expose to the Orchestrator the following functions of its own reservation application, as specified by **OMMT.Book**: *createOffers* to return offers for a specific trip, *postBookings* to create a booking based on a previously requested offer and *postFulfillments* to confirm the booking and trigger the fulfilment (the fulfilment may consist of a code to be input in a gate keypad or a barcode to be read by a reader).
Journey	If the parking service is not pre-paid, to communicate to the Orchestrator the check-in when the traveller's vehicle enters the parking and the check-out when vehicle exits, the Operator has to integrate in its own application the **OMMT.Sync** API exposed by the Orchestrator and consume the function *addAnnotation.*
Revenue Settlement	No implementation effort is required of the Operator. The Operator will receive periodically a detailed report from **OMMT.Share.**

Scenario 3

In this scenario, the Integrated Experience Model is either **One Purchase** or **One Media.** The parking service is charged by the Operator.

Step	What is required of the Operator
Booking	The Operator has to expose to the Orchestrator the following functions of its own reservation application, as specified by **OMMT.Book**: *createOffers* to return offers for a specific trip, *postBookings* to create a booking based on a previously requested offer and *postFulfillments* to confirm the booking and trigger the fulfilment (the fulfilment may consist of a code to be input in a gate keypad or a barcode to be read by a reader).
Journey	To enable the validation / inspection of the traveller, no implementation effort is required of the Operator, that can use its own validation / inspection systems.
Revenue Settlement	No implementation effort is required of the Operator. The Operator will receive periodically a detailed report from **OMMT.Share**.

11. Conclusions

Digital technologies and dematerialisation are transforming mobility. Customers expect integrated, seamless mobility experiences encompassing multiple providers and modes. But interoperability of diverse and fragmented processes and systems is a challenge.

OMMT is a one-stop-shop solution from UIC, ready for adoption. It provides a set of open digital specifications and services that accelerate and simplify the implementation and operation of key components of multimodal door-to-door mobility, including booking, validation, inspection, and revenue settlement among partners.

OMMT enables the rail industry to increase the cooperation with other modes of transport to provide efficient door-to-door mobility.

Version 1.0 – November 2025

Prepared by EY Advisory S.p.A. for the International Union of Railways