

Multimodal Mobility and MaaS made simple: the Overview Guide to OMMT



Contents

Executive Summary	3
1. The roadmap to door-to-door multimodal mobility and MaaS: rewards and challenges	4
2. What is OMMT and how it can help	6
3. OMMT Components	8
4. Enabling easy integrated customer experiences with OMMT	12
5. OMMT technology base	20
5. Roles in OMMT	22
6. Overview guide for rail undertakings	27
7. Overview guide for public transport operators	29
8. Overview guide for long-distance bus and ferry operators	31
9. Overview guide for vehicle sharing operators	33
10. Overview guide for taxi / ridehailing operators	35
11. Overview guide for parking operators	37
Annex – Understanding dynamic barcodes	38

Executive Summary

Providing integrated door-to-door mobility services requires cooperation among multiple transportation modes and service providers. Unfortunately, achieving the **interoperability and integration of the different technical systems presents a significant obstacle**.

OMMT (Open MultiModal Toolkit) is an **open framework** available to public and private transportation Service Providers that voluntarily agree to commercial partnerships to deliver seamless, end-to-end mobility experiences for customers. It is addressed to Public Transport Operators and to providers of Rail, Bus, Ferry, Vehicle Sharing, Parking and Ride-hailing services.

OMMT enables Service Providers to **bypass expensive and unique system integrations** that cannot be easily replicated. By adopting standardized, internationally recognized, secure, and tested specifications, Service Providers can speed up their entry to the market, lower the costs of implementation, and safeguard their investments.

OMMT is designed to **minimize the need to modify existing IT systems**. It does not necessarily call for new IT infrastructure investments. The model is adaptable and flexible, **supporting a variety of business models**. It enhances the user experience by offering various options such as booking in advance, purchasing digital tickets (either integrated or separate), as well as pay-as-you-go and account-based travel options.

OMMT also facilitates **international collaborations and cross-border travel**. It supports multiple models recognized by EU (single multimodal contract as well as combined and separate multimodal tickets). Its open nature is aligned with the EU MMTIS Directive. OMMT can help the realization of the EU vision of cross-Europe multimodal mobility.

OMMT bridges and **can work together with key mobility standards**, including Transmodel, OJP, NetEx, SIRI, GTFS and GBFS. In fact, it does not deal with journey planning nor with the exchange of timetables and fares. Its focus is on enabling the dynamic digital interactions that support dynamic offers, reservation of resources, secure validation and control, account-based travel, and fair sharing of revenues among providers.

This **Overview Guide is intended for managers and professionals** in the mobility sector, as well as for mobility authorities. It presents the concepts of OMMT in an accessible manner, minimizing technical jargon. The guide includes practical use case examples and tailored advice for different types of Service Providers.

Mobility authorities and industry stakeholders are encouraged to view OMMT as a valuable tool that can streamline and expedite the provision of integrated mobility services to the public.

1. The roadmap to door-to-door multimodal mobility and MaaS: rewards and challenges

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 (MaaS) 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 **dematerialisation** 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.

Traditional integrated multimodal offerings are usually 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.

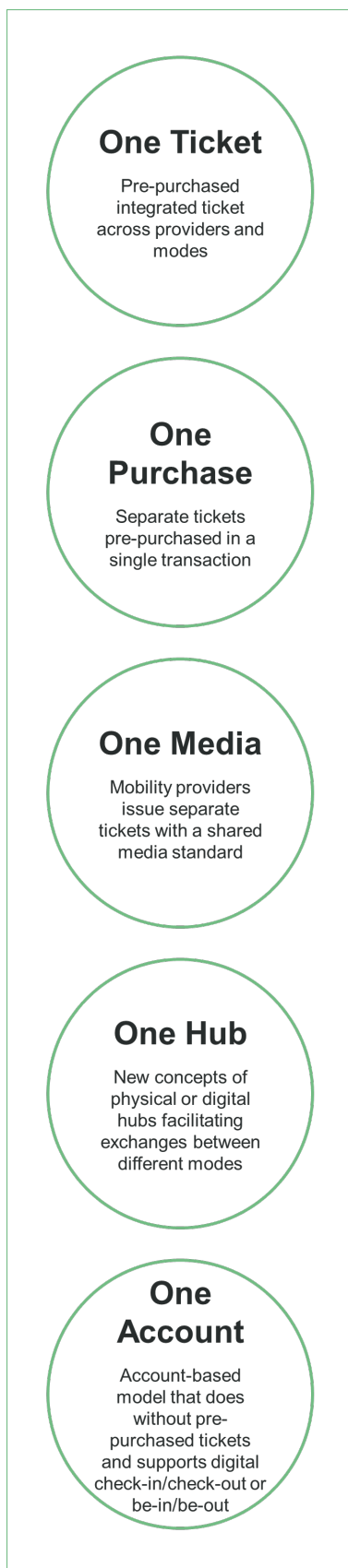
However, the advancement of Mobility as a Service (MaaS) and the broader concept of digitally enabled seamless door-to-door mobility have been hindered by **several obstacles**. These include:

- The need to align the objectives of various stakeholders and establish business models that are economically viable.
- The presence of disjointed and incompatible processes and technological systems.
- Lack of common technical specifications that span the complete range of door-to-door mobility services.

OMMT helps overcoming the above challenges.

It facilitates the integration of various transportation modes and service providers with minimal technological effort. Instead of prescribing a single business approach, OMMT is designed to be adaptable and versatile. This flexibility allows different entities to craft commercial agreements that are beneficial to all parties involved.

In fact, there are different ways to enable an integrated passenger experience, encompassing multiple mobility providers and modes. **Five main models** can be identified.



An outline of the five models follows.

One Ticket

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

One Purchase

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, for example a barcode, that is used as a fulfillment for allowing the passenger to access the mobility services.

One Hub

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

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).

Figure 1 - The five Integrated Experience models

2. What is OMMT and how it can help



The problem

In order to provide digital integrated door-to-door mobility services, multiple transportation modes and mobility operators have to cooperate. But the intent to cooperate is not enough. Achieving interoperability among processes and systems of the mobility operators is required and is often the key roadblock. Realising such interoperability requires **organisational cooperation in processes** based on joint agreements, as well as **enablers of technological interoperability**.

The exchange among operators of static data, such as timetables and fares, does not solve the problem. **Dynamic digital interactions are required** to enable, for example, dynamic offers and prices, reservation of seats, secure validation and control, account-based travel, fair sharing of revenues among providers.

OMMT: what is it?

OMMT provides a set of **open digital specifications and services** that enable digital interactions among Service Providers and simplify the implementation and operation of key components of multimodal door-to-door mobility, including booking, validation, inspection, and revenue settlement.

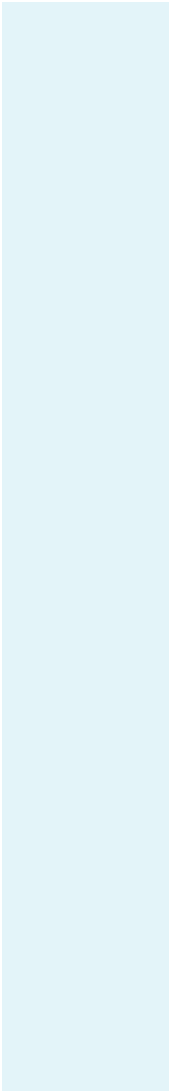
How OMMT helps

OMMT allows Service Providers to avoid costly, non-replicable, custom-fit integrations among their systems. Adopting **common, international, secure and proven specifications** reduces time-to-market, optimizes implementation costs and protects investments.

OMMT is designed to minimize (and in some cases eliminate) the effort required to adapt the IT systems. It does not necessarily require investments in new IT infrastructure. It is versatile and flexible. As described in the following, 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, OMMT is a **fully modular model**, which makes possible the adoption of individual components or of the full model. It is intended to cooperate with other solutions and industry standards.

Moreover, OMMT is grounded on specifications and solutions that have already been adopted internationally by mobility providers.

In summary, **OMMT benefits include:**

- Common, secure and scalable model that minimizes the effort required to Service Providers and protects investments.
- Support multiple types of integrated mobility experience and multiple MaaS delivery models.
- Simplify the design and implementation of multimodal / MaaS offerings through modular, integrated, standardised, scalable components.
- Facilitate the adoption by the mobility ecosystem through open specifications and simple implementation guidelines.

3. OMMT 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, respectively, for the following key capabilities required for providing digital door-to-door mobility services:

1. **OMMT.Plan** provides multi-modal location data and supports the exchange of timetables and fares.
2. **OMMT.Book** streamlines the process of creating and managing offers and orders that span across various service providers.
3. **OMMT.Move** grants customers a secure, digital representation of their travel rights, enabling them to utilize various mobility services while allowing providers to confirm and inspect service entitlements.
4. **OMMT.Sync** ensures the real-time exchange of updated information regarding order status and customer journey details among service providers.
5. **OMMT.Share** enables the fair distribution of revenues among service providers, with adjustments based on the actual service usage.

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** that 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

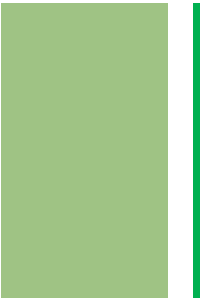
OMMT.Sync enables **sharing in real-time** updated information about the status of the order and of the customer journey among Service Providers, when needed.

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. Enabling easy integrated customer experiences with OMMT

In the following, some representative OMMT use cases are presented. For each use case, the customer journey is described. For each step in the customer journey, the enabling OMMT feature is indicated. The companies involved in the use cases have imaginary names.

Use Case 1: Pre-purchased multimodal trip

		Customer journey	Enabling OMMT feature
1		Sarah needs to reach a village that is served by a bus leaving from the rail station of the nearest town. She plans her journey through the mobile app Travel Planner of <i>Blue Rail</i> , that offers Sarah to book the entire journey at a convenient price, including, besides the train leg, reserved parking at the departure station c/o <i>Green Parking</i> and a reserved seat on connected bus from <i>Pink Bus</i> .	Through the OMMT.Book API, the <i>Blue Rail</i> booking application requests the availability and the price of the parking to the <i>Green Parking</i> booking application and of the bus to the <i>Pink Bus</i> booking application.
2		Sarah decides to book the entire trip and proceeds to pay it.	Through the OMMT.Book API, the <i>Blue Rail</i> booking application books the parking on <i>Green Parking</i> (which provides the <i>Blue Rail</i> booking app with a code that the customer should input on the keypad at the entrance) and the bus seat on the <i>Pink Bus</i> booking application (that generates a unique own booking reference and provides it to the <i>Blue Rail</i> booking app). The <i>Blue Rail</i> booking application sends the transaction data to OMMT.Share .
3		Sarah gets the confirmation of the trip on the <i>Blue Rail</i> mobile app, a dynamic barcode to be used for the train and the bus segment of her trip and the numerical code needed to enter the parking.	The <i>Blue Rail</i> booking application generates the dynamic barcode through the encoding function provided by OMMT.Move .

4		Sarah drives to the departure rail station and enters the parking by dialing the numerical code. She boards the <i>Blue Rail</i> train and shows the inspector the dynamic barcode on the mobile app.	The <i>Blue Rail</i> inspector checks the validity of the barcode by scanning it with a mobile device, that includes the decoding function provided by OMMT.Move.
5		After arriving at the connecting rail station, Sarah boards the reserved bus by showing the driver the dynamic barcode on the mobile app.	The <i>Pink Bus</i> driver checks the validity of the barcode by scanning it with a mobile device, that decodes the barcode through a function provided by OMMT.Move and retrieves the booking reference from the <i>Pink Bus</i> booking system.
6		At the end of the month, the involved Service Providers (<i>Blue Rail</i>, <i>Green Parking</i> and <i>Pink Bus</i>) receive a report with the owed balances and a detailed transaction log.	The report is generated by OMMT.Share.

Use Case 2: Account-Based Mobility

		Customer journey	Enabling OMMT feature
1		Mario travels occasionally for work reasons using public transport services by multiple providers, that have agreed to offer jointly Account-Based Mobility. Mario downloads the mobile app of one of the providers, <i>Red Bus & Metro</i> , opens a dedicated mobility account by providing his personal data and a payment method, and gets a dynamic barcode that identifies his account.	The <i>Red Bus&Metro</i> app generates the barcode identifying the account through OMMT.Move encoding function and sends the data of the new account to the OMMT.Sync service.
2		Mario boards a bus from <i>Brown Transit</i> that will take him to the metro station. Upon boarding, he checks-in by a swipe on the <i>Red Bus & Metro</i> app. When an inspector boards the bus, Mario shows the dynamic barcode on his mobile device.	The app logs the check-in event in the OMMT.Sync service. The <i>Brown Transit</i> inspector scans the account barcode and decodes it through the OMMT.Move decoding function. The inspection app accesses the OMMT.Sync service API and gets the confirmation that the account is valid and that the passenger has checked-in.
3		At the <i>Red Bus & Metro</i> metro station, Mario shows the barcode to a gate reader. The gate opens and Mario continues his trip.	The gate readers in the metro station download periodically the whitelist of valid accounts from the OMMT.Sync service. The gate reader decodes the barcode through the OMMT.Move decoding function and checks that the account is in the whitelist before opening the gate. The gate reader logs the event through the OMMT.Sync service.

4		At the end of the month, Mario receives via e-mail an account statement with the amount that will be charged, and the detailed list of mobility services used during the month.	The OMMT.Sync service has automatically communicated all the events to the OMMT.Share service. At the end of the month, the OMMT.Share service aggregates all the events related to Mario's account, invokes an algorithm agreed by the participating providers that computes the amount due by Mario (possibly applying fare capping rules) and how the revenue should be split among providers, and provides the results to <i>the Red Bus & Metro</i> customer application, that generates Mario's account statement.
5		At the end of the month, the involved Service Providers receive a report with the owed balances and a detailed transaction log.	The report is generated by OMMT.Share.

Use Case 3: Country-wide Digital Pass

		Customer journey	Enabling OMMT feature
1		In Alice's country, a country-wide digital pass is available, allowing unlimited usage of Public Transport services (excluding High Speed Rail) for a fixed monthly price. This involves dozens of PTOs around the country. Alice buys the country pass on the website of one of the PTOs, <i>Brown Mobility</i>, and receives a dynamic barcode that uniquely identifies the pass.	The <i>Brown Mobility</i> sales system generates the 2D-barcode through the OMMT.Move encoding function and sends the data of the new pass to the OMMT.Sync service.
2		Alice travels frequently around the country with Public Transport. She shows the dynamic barcode identifying her digital pass to gate readers in stations and to onboard inspectors.	The gate readers and the apps in the inspectors' mobile devices decode the barcode through the OMMT.Move decoding function and check the validity of the digital pass either locally, thanks to the security features of the dynamic barcode, or centrally through the OMMT.Sync service.

Note: In Use Case 3, also **OMMT.Share** could be leveraged, to record actual usage of mobility services by users and support a usage-based algorithm for splitting revenues and public fundings among operators. To collect reliable usage data, users should be enabled and incentivized to validate their pass each time they board a service. However, such an objective might be reached more easily through an Account-based Mobility approach (see Use Case 2), where the monthly cost is capped to the Digital Pass monthly amount.

Use Case 4: MaaS Mobility Package

		Customer journey	Enabling OMMT feature
1		Anna does not want to own a car and likes the idea of a single service catering to her mobility needs. She shops on the <i>Orange MaaS</i> website a personalised bundle of mobility services. It is a monthly package that includes a pass for Public Transport services provided by <i>Red Bus & Metro</i> , a 4-hour free allowance of <i>Pink Bikes</i> haring services and up to 4 city rides with the <i>Yellow Taxi & Ridehailing</i> company with a 20% discount. All for a convenient price.	Through the OMMT.Book API, the <i>Orange MaaS</i> sales system requests the <i>Red Bus & Metro</i> to issue a digital monthly pass in Anna's name and to provide the QR code in the proprietary format that is recognized by <i>Red Bus & Metro</i> validators and inspectors. It also requests <i>Pink Bikes</i> haring sales system to issue a 4-hour free coupon and to provide the numerical code that unlocks it when input in <i>Pink Bikes</i> haring app.
2		Upon payment of the monthly mobility package, Anna receives the QR code to be used to access Public Transport services provided by <i>Red Bus & Metro</i> , and the numerical code to be input in the <i>Pink Bikes</i> haring app.	Through the OMMT.Book API, the <i>Red Bus & Metro</i> QR code and the <i>Pink Bikes</i> haring numerical code are provided to the <i>Orange MaaS</i> sales system.
3		When traveling by Public Transport, Anna uses her pass as any other user. She is automatically entitled to 4 hours of shared bike <i>Pink Bikes</i> haring. One night she needs to book a taxi and does so from the <i>Orange MaaS</i> app, receiving an estimated price for the ride that takes into account the 20% discount.	Through the OMMT.Book API, the <i>Orange MaaS</i> requests to book a ride and its price from the <i>Yellow Taxi & Ridehailing</i> sales system. Upon confirmation from the customer, through the same API, the ride is booked.
4		At the end of the taxi ride, Anna gets from the <i>Orange MaaS</i> app a notice with the actual price of the ride, with the 20% discount, that will be charged to her account.	At the end of the ride, the <i>Yellow Taxi & Ridehailing</i> system confirms that the ride has taken place, with the actual price, to the OMMT.Sync service. Through OMMT.Sync , the <i>Orange MaaS</i> system and the OMMT.Share service are updated with this information.
5		At the end of the month, the involved Service Providers receive a report with the owed balances and a detailed transaction log.	The report is generated by OMMT.Share .

Use Case 5: Cross-border Multimodal Mobility

		Customer journey	Enabling OMMT feature
1		Adam and Aurora plan to spend the weekend visiting a beautiful town across the border and its surrounding region. When they book their high-speed train trip to their destination on <i>Blue Rail</i> website, they are offered with their trip a special 2-days pass that would allow them to move around freely in the foreign town and surrounding region using any local rail and Public Transport service.	<i>Blue Rail</i> has a commercial agreement with <i>Violet Express</i> , a rail company that operates in the destination country chosen by Adam and Aurora. Through the OMMT.Book API, each one can sell products offered by the other. <i>Violet Express</i> offers OMMT-enabled multimodal passes involving other PTOs in its home country.
2		Adam and Aurora decide to book the trip with the 2-day pass and proceed to pay it.	Through the OMMT.Book API, the <i>Blue Rail</i> booking application books the 2-day pass from the <i>Violet Express</i> booking application. The <i>Violet Express</i> booking application sends the pass data to the OMMT.Sync service.
3		Adam and Aurora each receive a barcode to be used for the train journey and for moving around during the weekend.	The <i>Blue Rail</i> booking application generates the barcode through the encoding function provided by OMMT.Move . Thanks to its flexible nature, it includes both contracts, covering the high-speed rail trip and the 2-day pass at destination.
4		Adam and Aurora board the <i>Blue Rail</i> high speed train and show the inspector the barcode.	The <i>Blue Rail</i> inspector checks the validity of the barcode by scanning it with a mobile device, that includes the decoding function provided by OMMT.Move .
5		During the weekend, Adam and Aurora use several buses and local trains and, when required, show the barcode to validators and inspectors.	Validation and inspection devices scan the barcode, decode it through the decoding function provided by OMMT.Move and check that the pass is valid through the OMMT.Sync API.

Use Case 6: Missed Connection

		Customer journey	Enabling OMMT feature
1		Lara is planning a vacation in a beautiful island, that she can reach by a <i>Blue Rail</i> long-distance train and <i>Purple Ferries</i> hydrofoil. The <i>Blue Rail</i> app offers her to book the entire journey at a convenient price.	Through the OMMT.Book API, the <i>Blue Rail</i> booking application requests availability of a seat and offer to <i>Purple Ferries</i> , which applies dynamic pricing to its services.
2		Lara decides to buy the entire trip and pays it. She receives a single barcode.	Through the OMMT.Book API, the <i>Blue Rail</i> booking application books the seat on the <i>Purple Ferries</i> hydrofoil and gets a booking reference number, that it encodes in the barcode, together with its own reference, through the OMMT.Move encoding function. The <i>Blue Rail</i> booking application sends the booking data to the OMMT.Sync service.
3		Lara spends a wonderful vacation in the island. When she returns, due to rough sea conditions, the <i>Purple Ferries</i> hydrofoil she is on is delayed by one hour. Lara informs the onboard agent that she has purchased a combined trip and will likely miss the connection with return long-distance train. The onboard agent instructs Lara to access in 5 minutes the <i>Blue Rail</i> app and change her reservation.	The <i>Purple Ferries</i> onboard agent app gets the reservation id from the barcode showed by Anna through the decoding function provided by OMMT.Move . Through the OMMT.Sync API, it records that the passenger is indeed on the booked vehicle and that it has a one-hour delay.
4		When Lara accesses the <i>Blue Rail</i> app, she finds a message saying: "Dear Lara, we are sorry that your vehicle is delayed. You can change your connected leg booking at no extra cost". Lara changes her <i>Blue Rail</i> reservation to a later train at no extra charge, although she had purchased an economy fare that did not allow changes.	According to the commercial agreement in place between <i>Blue Rail</i> and <i>Purple Ferries</i> , <i>Blue Rail</i> is responsible for missed connections. The <i>Blue Rail</i> booking application receives from OMMT.Sync the information that the customer is on a delayed segment and allows to customer to rebook without any surcharge.

5. OMMT technology base

OMMT is built upon a strategic selection of UIC specifications and services, targeting the core needs of digital multimodal mobility. Since OMMT does not require to implement fully the UIC specifications, the adoption complexity for transport service providers is greatly reduced, as explained in the following Overview guides.

OMMT.Plan technology base

OMMT.Plan leverages the MERITS service for the exchange of timetable data, including in GTFS, the DRTF fare distribution system, and the TSGA location API, built according to OSDM specs and supporting locations relevant to multiple transport modes.

Real world application example

MERITS has been for many years the most complete source of Europe-wide quality-checked integrated timetable data, used by railway undertakings and third parties in their information systems to support journey planning and booking



OMMT.Book technology base

OMMT.Book incorporates selected APIs derived from the OSDM (Open Sales and Distribution Model) specifications. OSDM is an international specification aimed at simplifying the booking process for customers.

Real world application example

Launched in 2024, the Sweden National Distribution System (NDS) developed by Samtrafiken, will allow customers to book tickets from more than 50 Swedish rail and bus operators, including the country's 21 independent regional Public Transport Authorities.

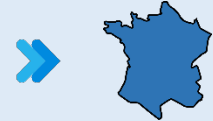


OMMT.Move technology base

OMMT.Move incorporates FCB (Flexible Content Barcode) v2, the latest, most advanced, and secure of the barcode specifications from UIC. It provides for flexible content that can be used for different transport modes and contract types. It can be validated/checked automatically, in both online and offline modes. It is part of the URT (Universal Rail Ticket), a standard layout to inform the passenger about key itinerary and contractual information. The UIC PKMW service supports the distribution of public keys.

Real world application example

In France, the Région Grand Est (over 5 million inhabitants and 36 different local transport networks) uses DOSIPAS to enable regional interoperable digital ticketing and seamless mobility across a range of mobility providers, with different ticketing systems.



OMMT.Sync technology base

OMMT.Sync leverages essential components of the eTCD specifications, that support the capability of controlling and annotating dematerialized tickets (e-tickets) that are not necessarily linked to a reservation. eTCD makes available API specifications and a cloud-based centralised common service for data exchange built according to the above specifications. The real-time sharing of data about delays leverages the RTMDE service by UIC.

Real world application example

Eurail offers customers a digital pass that allows them to travel by train to over 30,000 destinations in 33 European countries. eTCD enables data exchange with multiple providers. Customers can travel just by showing their Pass, which is securely inspected on board.



OMMT.Share technology base

OMMT.Share incorporates the NGRS (Next Generation Revenue Sharing) specifications. NGRS is a new concept developed by UIC, designed natively to support multimodal contexts.

5. Roles in OMMT

In the OMMT vision, multiple Service Providers collaborate to provide seamless integrated mobility experiences to customers. One entity is responsible for orchestrating the services, ie combining the services and presenting the end-to-end mobility experience to the customer. Such an entity can be one of the Service Providers. Multiple Service Providers can play this role within a partnership, depending on the customer. The retail function, ie sale and collection of payment, can be part of the orchestration or fulfilled by a separate entity.

Therefore, three essential roles are envisaged by OMMT: Service Provider, Orchestrator and Retailer

- **Service Providers** (SPs) are providers of passenger transportation and related services (e.g., Public Transport, Rail, Vehicle Sharing, Long-distance Bus, Ferry, Parking) that collaborate through a commercial agreement to provide an integrated experience to users, enabled by OMMT. The technology investments required to SPs to collaborate via OMMT are minimal, as described in the following sections, each dedicated to a type of provider.
- The **Orchestrator** is responsible for implementing a flow orchestrating multiple OMMT components and combining services of multiple SPs, to enable an integrated experience for the customer. The technology capabilities required to an Orchestrator depend on which integrated experience use cases are to be enabled and are generally more significant than for a participating SP.
- The **Retailer** interfaces the customer and collects payments.

Choice of terms

In the digital world, **service orchestration** refers to the coordination and management of multiple digital services to deliver a composite process. Similarly, in OMMT the term **Orchestrator** best describes the role related to the coordination of multiple OMMT components and services of multiple Providers, to enable an integrated experience for the customer. In other frameworks, a similar role may be defined **Distributor**, **Aggregator** or **Integrator**.

In some frameworks, the Service Provider role may be called Transportation Service Provider (**TSP**) or Mobility Service Provider (**MSP**).

Possible configurations

The flexibility of OMMT enables many different configurations. Some examples follow.

SP with Orchestrator role

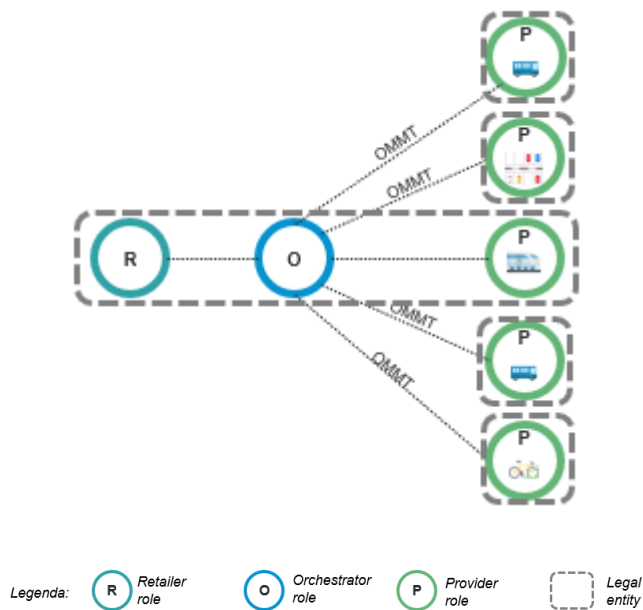


Figure 2 - SP with Orchestrator role

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

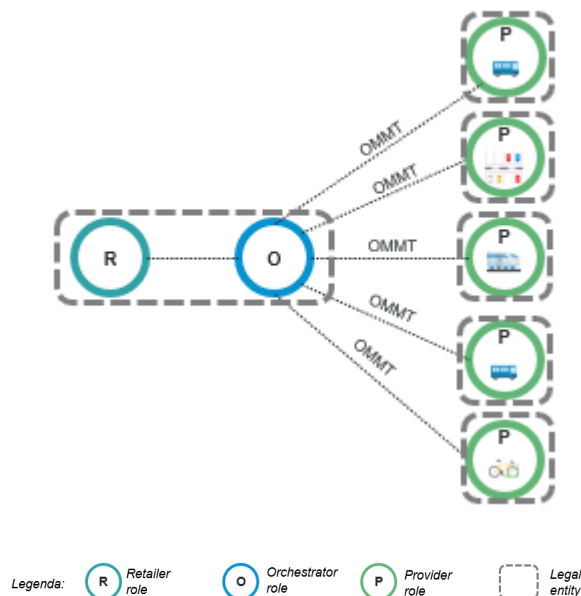


Figure 3 - Pure Orchestrator

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

Retailer separate from Orchestrator

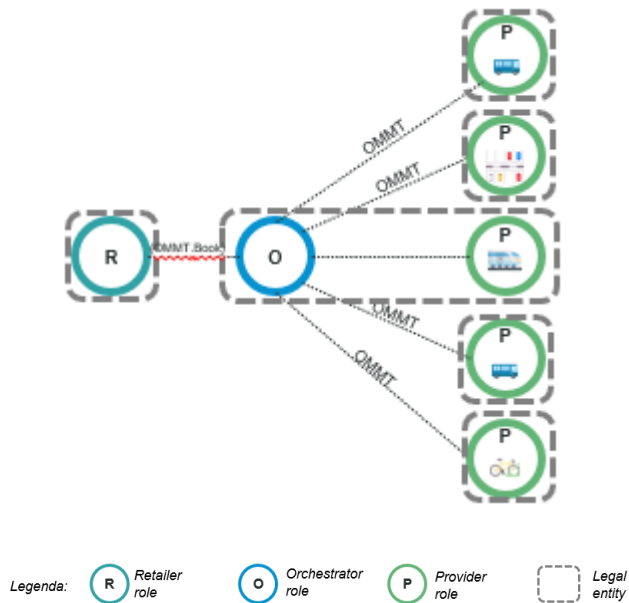


Figure 4 - Retailer separate from Orchestrator

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

Multiple Orchestrator Levels

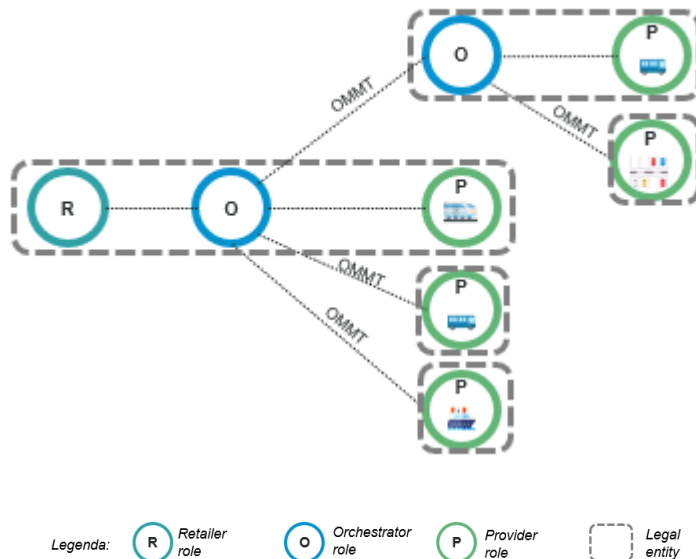


Figure 5 - Multiple Orchestrator levels

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

Multiple Orchestrators

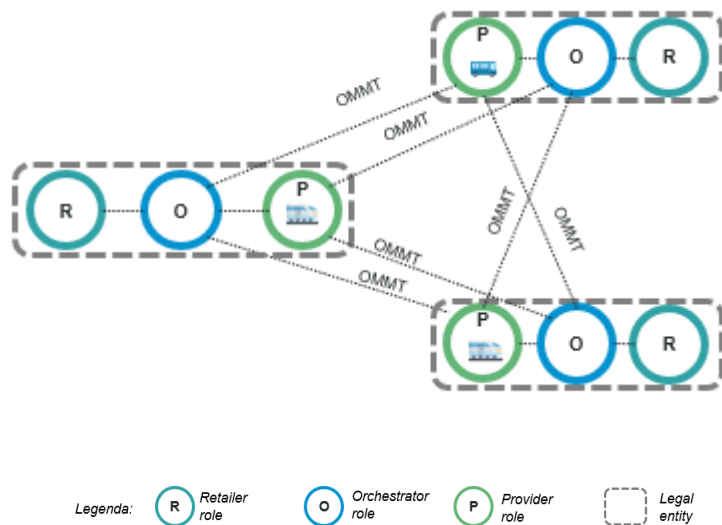


Figure 6 - Multiple Orchestrators

Technical Capabilities of Orchestrators

The technical capabilities required to the Orchestrator depend on the supported Use Cases. They include:

- Collect offers from SPs and issue orders via **OMMT.Book**. The typical steps are:
 - Translate a trip into a set of offer requests to SPs through the *createOffers* API service.
 - Have the customer choose a set of offers and collect payment (when acting as retailer).
 - Create bookings with SPs based on previously requested offers, through the *postBookings* API service.
 - Get the confirmation of the bookings and, when applicable, a fulfilment to be provided to the customer, through the *postFulfillments* API service.
- Generate the barcode with the trip or the account information through **OMMT.Move** and provide it to the customer (if dynamic barcodes are to be used, provide an app to the customer to generate and display dynamic barcodes).
- If applicable, provide the customer with all the additional SP-provided fulfilments, i.e. media required to access the mobility services such as specific barcodes, numerical codes, etc.
- Feed **OMMT.Sync** with order information, if applicable through the *postTicket* API service.

-
- Feed **OMMT.Share** with order information, if applicable, through the *postOrderTransaction* API service.
 - Feed **OMMT.Share** with information about the opening or closing of a customer account, if applicable, through the *accountStatusChange* API service.
 - Optionally, provide to SPs an app for controlling **OMMT.Move** barcodes.
 - Host the central **OMMT.Sync** service or ensure that partner SPs have access to a 3rd-party **OMMT.Sync** service.
 - Host the central **OMMT.Share** service or ensure that partner SPs have access to a 3rd-party **OMMT.Share** service.

Often, the interaction between the customer and the Orchestrator (or the Retailer when separate from the Orchestrator) will start with a journey planning step. In this case, the trips produced by the journey planner will be an input of the offer and order step. **OMMT does not address fully 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 **OMMT.Book** data model is consistent with the CEN Transmodel standard data model and the CEN Open API standard for distributed journey planning (OJP), but other data formats can also be adopted. **OMMT.Plan** may support the journey planning step, as it provides useful services for sharing timetable data and tariffs (particularly for railways, as many of them have already implemented interfaces to such services), and a service for retrieving location codes.

6. Overview guide for rail undertakings

A **Rail Undertaking (RU)**, in order to collaborate with other Service Providers (SPs) to provide an integrated mobility experience to customers through OMMT, should:

- Establish a commercial agreement with the Orchestrator.
- Make network and timetable information available to the Orchestrator. This can be done through any available format, for example via the UIC MERITS service (part of **OMMT.Plan**), or GTFS, a global industry standard.
- Enable its processes and system to be compatible with OMMT, as detailed below.



If the Rail Undertaking acts as Orchestrator, as well as SP, it should in addition ensure to have available the capabilities described in section 5. Such capabilities are typically already available to RUs that have aligned their systems to the components of the technology base of OMMT, as described in section 4.

What is required of a RU to be compatible with OMMT? The answer differs depending on whether the RU has already in place its own digital ticketing process or not.

A RU that has already in place its own digital ticketing process typically has the ability to issue digital tickets online (possibly including passes), provide a digital media that represents the ticket (such as a barcode), validate such media at gates in stations or on trains, and perform checks by inspectors on board. Such a RU should:

- Enhance its ticketing system with a digital sales channel, to issue digital tickets (which may require or not a reservation) in response to requests from Orchestrators. The digital channel should expose the following **OMMT.Book** API services: *createOffers* to returns offers for a specific request, *postBookings* to create a booking based on a previously requested offer, and *postFulfillments* to confirm the booking and provide Orchestrators with the digital media that the passenger has to use to access the RU mobility service.
- To enable the validation / inspection of the traveller, no implementation effort is required of the RU, that can use its existing validation / inspection systems. For the commercial agreements that require that actual usage data is reported to the Orchestrator (including for example *One Account* agreements), the application should record the event by using the **OMMT.Sync** *createAnnotation* API service.
- For the purpose of Revenue Settlement, no implementation effort is required of the RU. The RU will receive periodically a detailed report generated by the **OMMT.Share** service.

A RU that does not have in place its own digital ticketing process will typically rely on the digital ticketing capabilities of the Orchestrator, but needs to ensure the ability to check the validity of the digital tickets (or the digital accounts) issued by the Orchestrator:

- Integrate in its own validation / inspection application the decoding function that allows to decode the **OMMT.Move** barcode (an Android implementation is freely available).
- For additional security, the RU can retrieve online digital tickets / accounts by enabling its validation / inspection application to query the **OMMT.Sync** *getTicket* API service.
- All the above implementation efforts of the RU can be avoided if the Orchestrator provides the RU with a (white label) app that includes the above functionalities, and the PTO uses it in lieu of its own inspection app.
- For the purpose of Revenue Settlement, no implementation effort is required of the RU. The RU will receive periodically a detailed report generated by the **OMMT.Share** service.

7. Overview guide for public transport operators

A **Public Transport Operator (PTO)**, in order to collaborate with other Service Providers (SPs) to provide an integrated mobility experience to customers through OMMT, should:

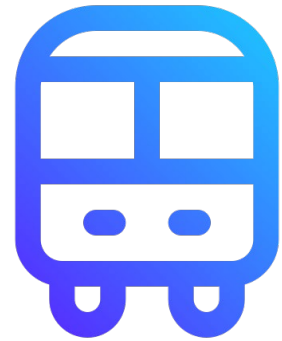
- Establish a commercial agreement with the Orchestrator.
- Make network and timetable information available to the Orchestrator. This can be done through any available format for exchanging static information (e.g., NetEx, the CEN standard adopted in Europe, or GTFS, a global industry standard) and/or real-time information (e.g., the CEN standard SIRI or the global industry standard GTFS-RT). The Orchestrator needs this information for Journey Planning, normally the first step in the sales process: it returns possible trips that fulfil the customer request. The full trip or part of the trip is included in the request for an offer to the public transport operator. However, OMMT supports also **non-trip-based offer requests**, such as multi ride products valid for multiple trips, single ride products 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), and reduction cards.
- Enable its processes and system to be compatible with OMMT, as detailed below.

If the PTO acts as Orchestrator, as well as SP, it should in addition develop the capabilities described in the section 5. This remark is valid also for the different types of SP that are the subject of the next sections.

What is required of a PTO to be compatible with OMMT? The answer differs depending on whether the PTO has already in place its own digital ticketing process or not.

A PTO that has already in place its own digital ticketing process typically has the ability to issue digital tickets online (possibly including passes), provide a digital media that represents the ticket (such as a barcode), validate such media at gates in stations or on buses, and perform checks by inspectors on board. Such a PTO should:

- Enhance its ticketing system with a digital sales channel, to issue digital tickets (which may require or not a reservation) in response to requests from Orchestrators. The digital channel should expose the following **OMMT.Book** API services: *createOffers* to returns offers for a specific request, *postBookings* to create a booking based on a previously requested offer, and *postFulfillments* to confirm the booking and provide Orchestrators with the digital media that the passenger has to use to access the PTO mobility services.
- For optimal support of non-trip-based offers, the PTO can expose the **OMMT.Book** *searchProducts* API. It supports multiple parameters are supported, e.g., begin of validity, regions / zones where the offers are valid, flexibility, list of products. It also supports searching for products by indicating product tags: this offers high flexibility, including even free-text requests (e.g., “I want to travel for two weeks in Germany”), by implementing AI search on the product catalog.



-
- To enable the validation / inspection of the traveller, no implementation effort is required of the PTO, that can use its existing validation / inspection systems. For the commercial agreements that require that actual usage data is reported to the Orchestrator (including for example *One Account* agreements), the application should record the event by using the **OMMT.Sync** *createAnnotation* API service.
 - For the purpose of Revenue Settlement, no implementation effort is required of the PTO. The PTO will receive periodically a detailed report generated by the **OMMT.Share** service.

A PTO that does not have in place its own digital ticketing process will typically rely on the digital ticketing capabilities of the Orchestrator, but needs to ensure the ability to check the validity of the digital tickets (or the digital accounts) issued by the Orchestrator:

- Integrate in its own validation / inspection application the decoding function that allows to decode the **OMMT.Move** barcode (an open Android implementation is freely available).
- For additional security, the PTO can retrieve online digital tickets / accounts by enabling its validation / inspection application to query the **OMMT.Sync** *getTicket* API service.
- All the above implementation efforts of the PTO can be avoided if the Orchestrator provides the PTO with a (white label) app that includes the above functionalities, and the PTO uses it in lieu of its own inspection app.
- For the purpose of Revenue Settlement, no implementation effort is required of the PTO. The PTO will receive periodically a detailed report generated by the **OMMT.Share** service.

8. Overview guide for long-distance bus and ferry operators

A **Long-distance Bus or Ferry Operator**, in order to collaborate with other Service Providers (SPs) to provide an integrated mobility experience to customers through OMMT, should:

- Establish a commercial agreement with the Orchestrator.
- Make network and timetable information available to the Orchestrator. This can be done through any available format for exchanging static information (e.g., NetEx, the CEN standard adopted in Europe, or GTFS, a global industry standard) and/or real-time information (e.g., the CEN standard SIRI or the global industry standard GTFS-RT).
- Enable its processes and system to be compatible with OMMT, as detailed below.



What is required of the Operator to be compatible with OMMT?

- Enhance its ticketing / booking system with a digital sales channel, to issue digital tickets (which may require or not a reservation) in response to requests from Orchestrators. The digital channel should expose the following **OMMT.Book** API services: *createOffers* to returns offers for a specific request, *postBookings* to create a booking based on a previously requested offer, and *postFulfillments* to confirm the booking and provide Orchestrators with the digital media that the passenger has to use to access the Operator's mobility services.
- Optionally, the Operator may use additional **OMMT.Book** API services to share a seat map with the Orchestrator and allow the customer to choose seat(s) on the map.
- To enable the validation / inspection of the traveller, no implementation effort is required of the Operator, that can use its existing validation / inspection systems. For the commercial agreements that require that actual usage data is reported to the Orchestrator (including for example *One Account* agreements), the application should record the event by using the **OMMT.Sync** *createAnnotation* API service.
- For the purpose of Revenue Settlement, no implementation effort is required of the Operator, that will receive periodically a detailed report generated by the **OMMT.Share** service.

It is also possible for Operator to rely on the digital ticketing capabilities of the Orchestrator. In this case, the Operator needs to ensure the ability to check the validity of the digital tickets (or the digital accounts) issued by the Orchestrator:

-
- Integrate in its own validation / inspection application the decoding function that allows to decode the **OMMT.Move** barcode (an open Android implementation is freely available).
 - For additional security, the Operator can retrieve online digital tickets / accounts by enabling its validation / inspection application to query the **OMMT.Sync** *getTicket* API service.
 - 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 uses it in lieu of its own inspection app.

9. Overview guide for vehicle sharing operators

A **Vehicle Sharing Operator**, that provides shared cars or free float / docked shared bikes or scooters, in order to collaborate with other Service Providers (SPs) to provide an integrated mobility experience to customers through OMMT, should:

- Establish a commercial agreement with the Orchestrator.
- Make available to the Orchestrator information about the location of available vehicles. This can be done through OMMT (see below) or through any other available format, including: NetEx, the CEN standard adopted in Europe; 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, an API Specification developed in Germany for shared cars..
- Enable its processes and system to be compatible with OMMT, as detailed below.



Vehicles covered by OMMT include Bicycle, Electric bicycle, Cargo _bicycle, Scooter, Self driving car.

What is required of the Operator to be compatible with OMMT?

- Enhance its booking system with a digital channel to book vehicles and/or offer coupons in response to requests from Orchestrators. The digital channel should expose the following **OMMT.Book** API services:
 - *createOffers* to returns offers for a specific request;
 - *postBookings* to create a booking based on a previously requested offer;
 - *postFulfillments* to confirm the booking and trigger the fulfilment. The fulfilment may consist of an unlock or discount code to be supplied to the traveller or may directly unlock the asset for immediate usage. Different models are supported, including prepaid, postpaid, auto-start usage, and auto-end usage.
- The operator can also expose the following **OMMT.Book** API services that specifically support on-demand mobility services, including vehicle sharing:
 - *getAvailabilitiesContinuousServices*, that allows the Orchestrator to provide as input geo position and types of vehicle (e.g., bicycle, electric bicycle, cargo bicycle, scooter, car) 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;
 - *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).

The above services support sharing of available pick-up places, service usage status management, service usage reporting, and usage tracking.

- For the commercial agreements that require that actual usage data is reported to the Orchestrator (including for example *One Account* agreements), the application should record the check-in when the vehicle is unlocked and the check-out when the vehicle is released by using the **OMMT.Sync** *createAnnotation* API service made available by the Orchestrator.
- For the purpose of Revenue Settlement, no implementation effort is required of the Operator, that will receive periodically a detailed report generated by the **OMMT.Share** service.

10. Overview guide for taxi / ridehailing operators

A **Taxi or Ridehailing Operator** in order to collaborate with other Service Providers (SPs) to provide an integrated mobility experience to customers through OMMT, should:

- Establish a commercial agreement with the Orchestrator.
- Enable its processes and system to be compatible with OMMT, as detailed below.

In this case, journey planning by the Orchestrator may not require interactions between the Orchestrator and the Operator, as journey planners usually can provide a reasonable estimate of the duration of a trip leg by taxi or ridehailing.



What is required of the Operator to be compatible with OMMT?

- Enhance its booking system with a digital channel to book vehicles and/or offer coupons in response to requests from Orchestrators. The digital channel should expose the following **OMMT.Book** API services:
 - *createOffers* to returns offers for a specific request;
 - *postBookings* to create a booking based on a previously requested offer;
 - *postFulfillments* to confirm the booking and trigger the fulfilment. The fulfilment may consist of a booking or discount code to be supplied to the traveller. Different models are supported, including prepaid and postpaid.
- The operator can also expose the following **OMMT.Book** API services that specifically support on-demand mobility services, including taxi and ridehailing:
 - *getAvailabilitiesContinuousServices*, that 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 the estimated vehicle arrival time at pick-up. The response can indicate the number and type of available vehicles and the pre-allocated time limit;
 - *getContinuousServiceUsageId*, that returns the on demand service usage;
 - *getContinuousServiceUsageId*, that allows to start or end the usage of an on demand service.
- The above services support sharing of available pick-up places, service usage status management, service usage reporting, and usage tracking

-
- For the commercial agreements that require that actual usage data is reported to the Orchestrator (in case the cost is not pre-paid and communicated at booking time), the Operator has to communicate that trip leg has ended and the cost of the trip leg. The Operator's application should communicate the information by using the **OMMT.Sync** *createAnnotation* API service made available by the Orchestrator.
 - For the purpose of Revenue Settlement, no implementation effort is required of the Operator, that will receive periodically a detailed report generated by the **OMMT.Share** service.

11. Overview guide for parking operators

A **Parking Operator**, in order to collaborate with other Service Providers (SPs) to provide an integrated mobility experience to customers through OMMT, should:

- Establish a commercial agreement with the Orchestrator.
- Make available to the Orchestrator basic static information about the parking facilities, such as locations and capacities, through any convenient format. An example is APDS from the Alliance for Parking Data Standards.
- Enable its processes and system to be compatible with OMMT, as detailed below.



What is required of the Operator to be compatible with OMMT?

- To enable reservation of the parking in advance, enhance its booking system with a digital channel to book vehicles in response to requests from Orchestrators. The digital channel should expose the following **OMMT.Book** API services: *createOffers* to returns offers for a specific request, *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.
- For the commercial agreements that require that actual usage data is reported to the Orchestrator (for example when the parking is not pre-paid), the application should record when the check-in when the customer's vehicle enters the parking and the check-out when vehicle exits by using the **OMMT.Sync** *createAnnotation* API service made available by the Orchestrator.
- For the purpose of Revenue Settlement, no implementation effort is required of the Operator, that will receive periodically a detailed report generated by the **OMMT.Share** service.

Annex – Understanding dynamic barcodes

Understanding DOSIPAS

The context

Mobile ticketing, in which the title to travel is represented on the user's smartphone by a **two-dimensional barcode**, has several advantages and has been adopted by many transport operators, in rail and TPL. However, in the **intermodal arena** it still has **limitations and drawbacks**:

1. The barcode format and content are **proprietary** and therefore valid for a single operator or consortium;
2. Interoperable products cannot be represented due to **lack of a common format**;
3. For a single door-to-door trip, the passenger is forced to make **multiple purchases** and **carry separate tickets**;
4. During validation and screening, **barcode verification is sometimes problematic** and is left out: online verification requires an active link, offline verification requires, to avoid fraud, a check of the passenger's identity.

DOSIPAS: main features

DOSIPAS (“DOuble Signed PAckage Structure”) is an **international standard** with the following characteristics:

1. Provides for **flexible content** (Flexible Content Barcode) that can be used for different transport modes and contract types, and in different contexts (TPL, regional, long-distance, etc.):
 - Single-ride tickets on rail, road or water;
 - Interoperable tickets that integrate different modes of transport;
 - Subscriptions;
 - Account-based ticketing.
2. It supports **two different security modes**:

- **Security in Data (SiD)**, in which barcode authenticity is guaranteed by data present in the barcode itself, through dual security signatures (static and dynamic), without requiring online checks;
- **Security in System (SiS)**, in which authenticity is guaranteed by data present in a central system accessible through a reference present in the barcode.

3. Enables a **wide set of use cases**, including:

- Offline validation or check;
- Online validation or check;
- Gate unlocking;
- Secure ticket identification for post-sale processes (e.g., refund).

DOSIPAS: advantages

DOSIPAS makes it possible to **overcome many of the current limitations** of barcodes in multimodal TPL mobility. The advantages are:

1. It is an **international interoperable standard**, not a proprietary format;
2. It **supports key use cases** in multimodal mobility;
3. It is secure and can be validated/checked **automatically**, in both online and offline modes (due to dual security signature);
4. It is **easy to implement**, as it provides:
 - Detailed specifications and documentation;
 - A centralized public repository/data service for static public keys;
 - An open public implementation of encryption/decryption functions that operators can freely integrate into their ticketing and control applications;
5. Offers a **proven solution** in the multimodal environment:
 - For example, in France, the Région Grand Est (over 5 mln inhabitants and 36 different local transport networks) uses DOSIPAS to enable regional interoperable digital ticketing.

DOSIPAS: how it ensures safety in case of offline control

Securities protected by SiD (Security in Data) have some **limitations** in terms of **security**. When a ticket is copied, it can also be used as a valid ticket. To avoid this kind of abuse, tickets are often **personalized**. This personalization makes **ticket checking more complex**: not only must the ticket be checked, but also an official document (such as an ID or regular customer card) of the passenger. A good alternative to add **extra-security** features to avoid copying is to add a dynamic part in the data that are translated into barcodes and displayed on the smartphone. This involves the creation of a so-called **Dynamic Barcode**.

DOSIPAS: how dual signatures ensure security

DOSIPAS supports **256-bit key pairs** (public/private). The issuing company uses a **static key pair**: the private key is used to sign the ticket, the public key (to be renewed every 18 months) is made available on a public repository. The app on the passenger's device generates a **dynamic key pair**. The public key is added to the ticket data and generates a secure static signature (since it is protected by the company's private key). The private key is used to sign the dynamic data and generate the barcode. The controller app **decodes the barcode** through the company's static key (available on repository) and the dynamic static key extracted from the barcode itself.

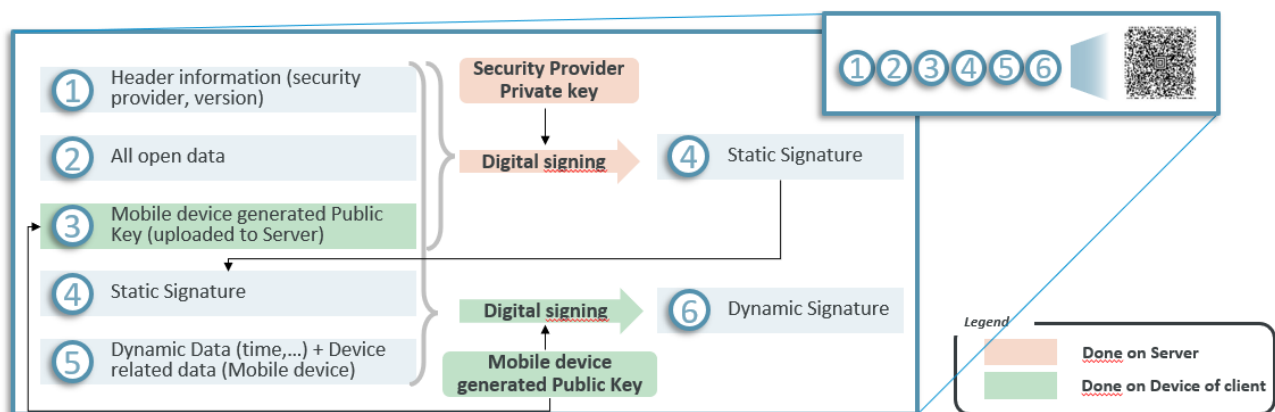


Figure 7 – DOSIPAS Process

RESOURCES

Together with this Overview Guide, UIC makes available:

- OMMT Essential Guide, a concise, high-level document summarising the key concepts contained in the Overview Guide, that allows to quickly understand what is OMMT and how it can help.
- OMMT Technical Guidelines, including an introduction, a technical description and guidelines for OMMT adoption, customized for each type of Service Provider (Rail, Public Transport, Bus, Ferry, Bike / Scooter / Car Sharing, Taxi / Ride-hailing, Parking), that helps to evaluate the effort required for adopting OMMT and the benefits it unlocks.
- Detailed, in-depth technical specifications, with all the information needed for implementation.

For further information please contact: d2d@uic.org