Interreg
Alpine Space
LinkingAlps
European Regional Development Fund

EUROPEAN UNION

Requirements Document

Incl. D.T1.3.1.-Specification of the System Architecture & D.T1.4.1.-Specification of the Active System (Short Report)

Version: 1.1

Date: 30.06.2021

Dissemination level: Public

WP: WP T1.3 and 1.4 – Specification of the System Architecture & Specification of the Active System

Authors: all implementing partners

Document Information

Authors / Document responsible

Name	PartnerNo	Organisation
Christoph Lucas	PP12	SBB
Matthias Günter	PP12	SBB
Katharina Helfert	LP / PP1	ATE
All OJP implementers		ALL

Peer-Reviewer

Name	PartnerNo	Organisation
Andreas Partusch	PP5	VAO
Bettina Neuhäuser	PP5	VAO
Daniela Gamper	LP / PP1	ATE
Andreas Glauser	PP12	SBB

Document History

Date	Version	Summary of changes	Dissemination
23.11.2020	V0.8	• First draft of Deliverable based on Confluence status 23/11/2020 (ATE) • Restructuring of content based on partner feedback (ATE) and to ensure a logic structure of document (ATE)	Project partners
01.06.2021	V0.9	Final version in discussion	Peer Reviewers
30.06.2021	V1.0	Reviewed version was published	Peer Reviewers

Interreg
Alpine Space
LinkingAlps
European Regional Development Fund

Preface

Acronym:	LinkingAlps
Title:	Innovative tools and strategies for linking mobility information services in a decarbonised Alpine Space
Project number:	740
Start Date:	01-10-2019
End Date:	31-06-2022
Call number:	4th call
Priority:	Priority 2 - Low Carbon Alpine Space
Specific objective:	SO2.2 - Increase options for low carbon mobility and transport

Abbreviations & Glossary

Abbreviation	Definition
ATE	AUSTRIATECH
AEV	Transport and Energy agency Canton Grison
ARIA	Regional Agency for Innovation and Purchasing Ltd
BLIC	Consulting company for control, information and computer technology GmbH
Cerema	Centre d'Etudes et d'expertise sur les Risques, l'Environnement, la Mobilité et l'Aménagement - Centre For Studies and Expertise on Risks, Environment, Mobility, and Urban and Country planning
CMTo	Metropolitan City of Turin
FoT	Federal Office of Transport
LINKS	LINKS Foundation - Leading Innovation & Knowledge for Society
NAP	National Access Point
RRA-LUR	Regional Development Agency of the Ljubljana Urban Region
SBB	SBB Swiss Federal Railways
STA	South Tyrolean Transport Structures
UM-FGPA	University of Maribor
VAO	Traffic Information Austria
VTG	Verkehrsverbund Tirol GesmbH

AS (Active system)	The active system integrates the routing information from several local journey planners to a combined seamless route. It is composed of a passive system and a Distributing system. It communicates through an OJP interface. It's a journey planning engine with OJP capabilities. Via the distributing system it is able to detect journeys through adjacent or remote regions and able to create OJP Trip Composition (alias OJP Router).
Adjacent region	Region which is adjacent to the local region and has its own "local" journey planning systems.
Adjacent system	Alias for neighboring system; participating System of an adjacent region.
DS (Distributing system)	System that distributes journey planning enquiries to other systems. It sends the request for journey-parts through areas to the corresponding passive servers, receives the responses and is able to create OJP Trip Compositions. It has the knowledge about gazetteers and is able to collect information about exchange points for the whole system.
EU (End user)	User of an "end user application". It is a person asking for journey planning information by using an end user application. It is the enquirer of a journey plan with a start, an end point and some travel preferences.
End user application (in this document, the term 'End user service' is used synonymously for 'End user application')	It is the application used by the end user to have access to JP information generated by the Distributed Journey Planning Service (DRJP). It can be a third party application connecting by OJP interface to a Participating system or the User Interface participating system. The provider of the end user applications are the, so called, "OJP users" in the LinkingAlps project.
Enquierer	The end user asking for information.
Enquirer system	Alias home system.
Estimated data	A predicted arrival or departure time of a particular means of transport at a particular stop. In case of real time data it can change several times during the journey.
Exchange point	Stop or stations where the trip leg of one system is connected to the trunk leg of another system. This includes regional stops which match with stops for long distance or regional stops from adjacent regions. Exchange points are mainly but not exclusively located at borders and in bigger cities.
Exchange point data base	A repository, a view on a database or a service that is able to list the relevant exchange points of the distributed service. It can be a static system-wide database or be generated dynamically with requests for exchange points to the responding services.
Gazetteer	directory of common objects across the local journey planner systems and its system borders. It enables the active system to find the passive system for all geolocations (stops, stations, POIs, address etc.). The gazetteer acts system-wide.

Home system	The Participating system called by the end user application. It is the system that take care of the end user travel information request and provide an answer.
Journey	A movement of a traveller from a start point to an end point by using one or more transport modes.
Journey Planner (JP)	A system that is calculating the journey for a given request. It is able to accept requests directly from end-user services. It is a generalization of OJP Router and OJP responder.
Journey Planning System (JPS)	Alias for Journey Planner.
Local Journey Planner (LJP)	A system with a routing engine and access to multimodal data with a particular local, regional or national coverage; "local" underlines its focus on a specific coverage that is limited. LJPs have no transregional (or distributed) OJP routing capabilities.
Local region	The territory covered the journey planner / home-system, which can plan trips itself without information from other systems.
Long distance traffic	Crosscountry interchange. Supported by trains and crosscountry travelling coaches.
Long distance transport connection	The trunk legs of the routes that connect at least 2 OJP systems. They are used to connect two neighboring or remote systems. Exchange points are defined along the trunk leg which define all the neighboring systems.
Neighbouring system	Alias for adjacent system.
OJP Implementer	Travel information service provider that is implementing an OJP service exchange (in most cases on the back-end system of an end user service).
OJP Interface	Application Programming Interface (API) based on CEN/TS 2017: OpenAPI for distributed journey planning and specified in D.T1.5.1 Specification of the API interface (including a LinkingAlps OJP Profile).
OJP Responder	Alias for passive system.
OJP Router	Alias for active System.
OJP User	End-user service provider that is using OJP services from local JPs to provide an end-user service
Open Journey Planning (OJP)	Standard for communication for distributed journey planning.(CEN/TS 17118:2017)
Participating system	A local journey planner being part of the OJP system architecture and the appropriate OJP service
Passive system	A Local journey planner (LJP) with a OJP interface (API) being able to respond to requests from distributing systems. It is an information source within the system without distributed journey planning capabilities. It communicates through an OJP interface as a responding system. alias OJP responder, responding system.
Public transport services	Service that allows people to travel. The service is for public usage.

Real time data	The real time of a particular means of transport at a particular stop; only sent after the arrival/depature of the vehicle at a particular stop.
Remote region	Region which is not adjacent to the local region. A remote region is covered by a local LJP.
Remote system	Participating system of a remote region .
Responding system	The generalized term for a system that responds to questions from the distributing system
Schedule data	planned data for public transport services
Server	Program that provides special services that can be used by other programs.
Service	Technical, self-sufficient unit that bundles related functionalities into a complex of topics and makes them available via a clearly defined interface.
System	Delimitable "structure" consisting of various components which can be regarded as a common whole due to certain ordered relationships between them.
Trip	Alias Journey.
Trip leg	The local part of a trip which is calculated by a single Local Journey planning system.
Trunk leg	The “trunk” leg are long-distance transport connections that interlink Journey planning systems
Linking Alps Distributed Journey Planning Service (DRJP)	Is a network of existing local, regional or national travel information services (routing platforms) that collaborate on the basis of CEN OJP exchange interface (CEN/TS 2017: openAPI for distributed journey planning) in order to exchange travel information and routing results. A web-based communication network between the participating systems needs to be established as the systems are physically remote. A universal common interface for exchanging requests between the participating services needs to be specified and implemented at all participating systems.
OJP Profile	The LinkingAlps OJP Profile aims to define a specific subset of (XML) data elements following a clearly arranged structure in accordance with the OJP standard (CEN/TS 17118:2017) and defined using XML schemas. The schemas include all functionalities required for an OJP interface in order to enable communication with the LinkingAlps distributed journey planning system. In this sense the LinkingAlps OJP Profile defines the content and the structure of the information content as well as the physical exchange format.

List of Figures

List of Tables

Table of Contents

1 Introduction

1.1 Purpose of the requirements document

The document contains the technical requirements for developing the LinkingAlps distributed journey planning system, which is a network of existing travel information service providers.

- The document defines the system components and its tasks and lays down the technical requirements for the technical implementation of the distributed OJP service in the Alpine region.

In this sense this document is the specification document for the technical architecture (and related aspects), as well as for the active and passive systems. The full technical specifications will be completed with deliverables D.T.1.5.1, which covers the specification of the interface (OJP profile) separately.

This requirements and definitions within this document have been developed jointly by the OJP implementers involved in LinkingAlps (ARIA, 5T, SBB, STA, VAO and LUR/UM-FGA for the Slovenian implementation). For the definition of the requirements the collaboration tool CONFLUENCE was used. The voting on the requirements took place unanimously during the bi-weekly calls between March 2020 – November 2020. The requirements concerning only the active systems have been agreed upon the active systems (SBB, STA, VAO).

This is the version 1.0 of the requirements document, which will be updated (into version 2.0) after the implementation and testing phase in WPT2, which takes place between November 2020 – June 2021.

This document covers the technical requirements for the implementation of the LinkingAlps distributed journey planning system.

Chapter 5 summarizes the organisational requirements that came up during the discussion. The concrete organisational architecture is handled in WPT3. Therefore, all requirements mentioned in this chapter are explained and covered in detail in deliverables D.T.3.2.1. (Organisational architecture).

This document is closely linked to deliverable D.T1.5.1. LinkingAlps OJP profile, which contains the actual specification of the OJP API used within LinkingAlps.

For the actual implementation, this is the necessary “implementation package”. Future implementers have to use “Requirements document”, the “LinkingAlps OJP Profile” and the “Overall Organisational Architecture” for an appropriate implementation.

1.2 Relevance of chapters for implementation of passive vs. active systems

While the upcoming chapters provide an overview of the OJP implementation in LinkingAlps in the following main chapters the technical requirements for the implementation are laid down in detail.

In principle the requirements as described in this document can be divided into the relevance for the implementation of the parts related to the active systems, other parts tackle the implementation of the passive systems specifically, while others imply requirements for the integration into end user services.

In the Annex to this document you will find a complete list of all requirements including this differentiation on the level of the individual requirements. The following table should just provide a rough overview, on the main topics tackled and the relevance of the technical requirements described in these chapters.

Number of Chapter	Chapter	Relevance for active system	Relevance for passive systems	Relevance for end user application
2	**System Overview**	no specific requirements defined		
2.1.	General considerations for the distributed journey planning	no specific requirements defined		
2.2.	System environment	no specific requirements defined		
2.3.	System architecture overview	no specific requirements defined		
3	**Description of the main components (incl. technical requirements)**	no specific requirements defined		
3.1.	Overview of components	no specific requirements defined		
3.2.	End user application (client)	x	x	x
3.3.	OJP Responder (alias Passive Systems - Responding system)	x	x	
3.4.	OJP router (alias Active System)	x		
3.5.	Distributing system	x		
3.6.	OJP Interface (OJP API based on a LinkingAlps OJP Profile)	x	x	
3.7.	System-wide gazetteers	x	x	
3.8.	Exchange Point Database or Service	x	x	
4	**Interactions**	no specific requirements defined		
4.1.	Interaction of OJP router and OJP responder	no specific requirements defined		
4.1.1.	Content and physical exchange of the openAPI/OJP interface	x	x	
4.1.2.	Handling of long-distance transport (to calculate the "trunk" leg) & exchange points for trunk leg calculation	x	x	
4.1.3.	Handling of exchange points & relevant requests	x	x	
4.1.4.	Supported requests, flow and sequence – demanded by the OJP routing and orchestration of messages	x	x	
4.2.	Interaction of OJP router and end user application	x		x
4.3.	Authorisation, authentication and compression of the network	x	x	
4.4.	Service and data quality	x	x	

Table 1: Overview of implementation relevant chapters

1.3 Verbal formulation of requirements in this document

In this document, the following verbal forms are used to indicate requirements: *Shall / Shall not*
Recommendations shall be indicated by the verbal forms: *Should / Should not*
Permissions shall be indicated by the verbal forms: *May*

1.4 Implementation phases

To ensure the necessary feedback loops between “requirements definition” and the actual implementation the work plan foresees the following phases and updates of the requirements document:

Requirements definition phase - (Requirements document versions 0.X – 1.0), WPT1&WPT3 (03/2020 – 11/2020)

This phase covers the actual technical discussion and development of the technical (and organisational) requirements jointly by the OJP implementers. For the joint development of the specifications, the collaboration tool "Confluence" was used in order to allow a joint development over all OJP implementers. The results of this phase are summarized in this document. The updated implementation plan foresees the finalisation of the specifications in version V.1.0 in 11/2020.

The activities of this phase related to the technical requirements are covered mainly by WPT1. The organisational requirements, which are closely connected to the technical requirements, are covered mainly by WPT3, but will often discussed commonly.

Implementation and Testing phase (Specification version 1.0), WPT2 (11/2020 – 06/2021)

Then there will be a "spec freeze" and implementation starts at all distributed OJP implementers according to the specification V.1.0. A comprehensive testing and implementation phase is planned from 11/2020 until 06/2021 for active and passive systems. In this phase implementation variants/differences will be identified in the integration phase of the API services at the distributing system. These variants/differences need to be evaluated and possibly removed during the integration and testing phase.

This phase is consequently covered by WPT2.

Change phase (Specifications version 1.X), WPT1 (10/2020 - 06/2021)

Occurring variants/differences mean that the requirements document V1.0 needs to be adapted. A change process for the specification is needed. In this phase of change the same working and alignment process apply as in the requirements phase (alignment of all OJP implementers is needed). Solutions need to be worked out and jointly agreed. The results from testing requires the specification V.1.0 to be adapted into a V2.0 that allows a fully interoperable implementation of the system components.

The adaption of the requirements document (and the D.T.1.5.1. OJP Profile) are part of the activities in WPT1.

Implementation of final specification (Specification version 2.0), WPT2 (06/2021 – 12/2021)

In this phase, all deviations in the local OJP services at all passive and active systems are removed according to Specification V2.0.

Evaluation phase, WPT2 (01/2022 – 06/2022)

Extensive testing phase with focus on technical functionality and specific use cases. Service quality and service functionality are further focal points. The tests are carried out until the service operates unconditionally.

1.5 System scope

1.5.1 System name

The LinkingAlps distributed journey planning system is a network of existing travel information service providers. In the LinkingAlps application form (part of the LinkingAlps Subsidy Contract) the LinkingAlps distributed system equals Output O.T2.1 Transnational mobility information service that is based on a decentralised network of linked journey planners.
The Output is described as "*An end-user service that provide seamless, transnational, multimodal trip planning in a single transaction (multilingual) and ease access to low-carbon mobility options (SO2) based on a novel linked decentralised (distributed) system (backends*)."

The name used in this document is LinkingAlps distributed journey planning system (LinkingAlps DJP). The mentioned backends are the participating active and passive systems.

1.5.2 System coverage

The mentioned inter-linked backends are the participating travel information services in the project, which are mentioned in table below.

Project Partner	**Travel information service**
VAO (AT)	Traffic Information Austria
SBB-SKI (CH)	Open Data Platform Swiss Public Transport
STA (IT)	South Tyrolian Journey Planner
ARIA (IT)	Muoversi Lombardia (Lombardy trip planner)
CMTo/5T	Muoversi Piemonte (Piemonte trip planner)
RRA-LUR/UM-FGPA	AtoB Slovenia

Table 2: Participating systems (back ends)

The total coverage of the LinkingAlps DJP equals the sum of the coverages of the participating services.

1.5.3 Key characteristics of the system

Due to the learnings from past project and system architectures to establish cross-operator travel information or EU-wide travel information service, the LinkingAlps distributed journey planning system is characterised by following key characteristics and fulfil following key tasks:

Compliance with Delegated Regulation 2017/1926

The objective of the European Commission regarding the provision of multimodal travel information services are major premises for LinkingAlps. The compliance with the Del. Reg. 2017/1926 supplementing the EU ITS Directive is a key pre-condition.

The network of journey planners will exchange information via OJP interface fully compliant in line with the Technical Specification CEN/TC 17118:2018 for "Open API for Distributed Journey Planning". The Delegated Regulation 2017/1926 of the ITS Directive envisaged the linking of services through that CEN/TC 17118:2018. The compliance with the Delegated Regulation and its provisions on European standards and technical specifications is a precondition. The key task of the system is to enable linking of services through the given provisions of the Delegated Regulation.

Decentralised architecture

Past initiatives to establish EU-wide travel information services by establish a central system architecture failed. This is because a centralised pool for travel data means to have a neutral pool operator that is interpreting the data. Because of missing trust on the neutrality on the central service provider and conflicting interests of data providers (source systems) and pool operator, as well as missing viable organisational models for a central European travel information service, the centralised approaches have not been successfully. Considering the past, the EC is regarding "linking of services" with a decentralised organisation of information provision as a key enabler for seamless travel information.

In the LinkingAlps decentralised and distributed journey planning system, a network of journey planners (participating systems) collaborate to compute journeys over a wide area beyond the domestic coverage of each planner. Participating systems are considered as either regional or national journey planning systems taking part in the decentralised, distributed system. The key task of the system is to enable information exchange between the participating services vice-versa. A standardized interface needs to be developed that allows the peer-to-peer exchange through APIs. This decentralized approach allows the journey planners to keep the sovereignty over their data and the interpretation of the data in their routing.

Nevertheless, this decentralised and distributed journey planning system needs some parts in the architecture, which are managed and handled system wide. This makes exchange of supporting data necessary. Considering performance issues, the exchange of data might be relevant. The architecture will be as distributed as possible, but with consideration of an operational, performant service. That means that the OJP interface will be designed in a way that it supports a high level of distribution, but exchange and integration of supporting data (like exchange points, gazetteers, long-distance transport) is not regarded as contradiction to distributed journey planning approach.

Within LinkingAlps both approaches will be considered and demonstrated, even if the core of the project is the fully decentralised, distributed system. Mainly triggered by the performance aspect SBB will set up a partly pooled approach, integrating e.g. timetable data for the whole covered network. The other partners, like STA and VAO, will follow the fully, distributed approach. The evaluation will show the differences in the implementation of both approaches, as well as differences regarding the performance.

Federated system

The architecture should follow as much as possible a federated approach. The system must provide access to several autonomous information sources without copying their data. It must integrate multiple data sources, in which the sources themselves remain unchanged. A virtual information integration or data federation is envisaged. The system must facilitate the combination of individual systems that retain their respective autonomy and sovereignty. In the case of the travel information services the sovereignty is understood as keeping the responsibility over the data maintenance and quality and interpretation of the data (e.g. in the routing algorithm).

Scalability of the system architecture

The system architecture is considered scalable. That means that the system must provide the opportunity to extend and to include further OJP services.

Operationalisation of the service and usability

One major premises for the system architecture (and the whole LinkingAlps project) is to produce an operational product service after the end of the project. Therefore, the operationalisation of the service as well as the usability are major requirements. The system architecture needs to be accompanied by the operational model and the definition of roles, tasks and responsibilities in the network. WPT3 is defining a complementary operational model and the processes supporting an operational service.

The LinkingAlps project's objective is to provide at least one end user service and bring the results to the citizens of the Alpine Space. Therefore, the usability of the service is of utmost importance. So the content and the opportunities (functionality according to user needs) of the information service, as well as the performance and the security are key requirements for the system.

1.5.4 Expected advantages of the system

The expected advantages of the service are depending on the different stakeholder groups involved (see also chapter 1.6. on more details on the stakeholder groups). In the following the major groups and their advantages are summarized.

<u>For OJP implementers (=OJP service providers):</u>

- Extension of the coverage of the existing travel information services by integrating travel information from other sources dynamically by using APIs
- Provision of seamless travel information to the passengers without comprehensive data exchange processes
- The actuality of the information is always according to the source system and no inconsistencies arise
- The route calculation is always carried out by the actual source system and is therefore only in the hands of the routing provider - no misinterpretations possible
- Data provider can also be data user at the same time (passive or active role is possible), therefore a utilization in own service is possible

For OJP users (=end user application providers):

- The providers of end user services can use an "ecosystem" of OJP services that are taking over the role of an integrator of travel information with their expert role in the transport operator domain.
- They can use trusted and reliable information to build up special end user services
- They can focus on the end user application side in the development and do not need expert knowledge in the whole travel information production chain.

For end users (=travellers):

- improved accessibility to travel information for the passengers and increased acceptance of public transport and alternative mobility
- extended geographical coverage of the well-known end user application

1.6 Stakeholders

The requirements document was specifically formulated for the set of stakeholders involved in the development of the LinkingAlps distributed journey planning system. Furthermore, the requirements and specifications within this document provide also an orientation on minimum system requirements for future OJP-adopters, who are interested to join in the LinkingAlps distributed journey planning system

Overall, stakeholders involved in the development and implementation of the LinkingAlps distributed journey planning system, cover **actors, who provide the front-end (such as OJP users)** and the **back-end (such as OJP implementers)** of the fully distributed journey planning system. Furthermore, **supporting actors (such as research, academia or solution developers)** who provide insights into existing system configurations, but also drive the technical innovation of the OJP-ecosystem and **facilitating actors (such as ministries, policy makers and decision makers)**, who provide input and establish framework conditions to enable the implementation of the OJP-based LinkingAlps system.

Due to the key focus of this document, to provide the foundation for a transferable, technological minimum readiness of participating systems, the **herein formulated requirements and specifications are mainly directed at back-end actors of the LinkingAlps distributed journey planning system, thus, OJP implementers.** However, this section will outline the relevance of the requirements document for all involved stakeholders as well as discuss their roles and potential interests in regards to the OJP-ecosystem.

Stakeholder type		**Actors in LinkingAlps**
Front-End	OJP User	STA and future project external end user service providers
Back-End	OJP Implementers	ARIA, SBB, CMTo/5T, LUR, STA, VAO, and future OJP implementers
Supporting	Academia, researcher, standardisation	Tool and solution developers, like UM-FGPA, CEREMA, LINKS

Facilitating	Decision makers, policy makers	BMK, FoT, MIS, CMTo, ATE, Regional governments and other policy-related decision makers.

Table 3: Overview of LinkingAlps stakeholders

In the following main stakeholder types are described more in detail.

1.6.1 End-User

End-Users are the central actor within the LinkingAlps distributed journey planning system and are defined as a public person (traveller) asking for information at an End-User-Application to plan a journey.
The two key End-User target groups within the LinkingAlps system are typically cross-regional or transnational commuter and tourists. Both groups show a growing demand for accurate and multimodal travel information across borders and regions. Whenever possible they want real-time-data.

While commuters, consider real-time notifications on incidents on their daily routes as essential service feature, tourists benefit from a one-point access to generate seamless, transnational and multimodal trip plans, which incorporate low-emission transport modes also in unfamiliar territory.

Within the requirements document End-User hold only an informative role. However, End-User preferences, usage behaviour and consumption behaviour depict an orientation point of the planning and implementation of the whole system.

The End-User is the actor in the use cases and the enquirer of a trip plan.

OJP services enabled End-Users to use their familiar local journey planning application (which is provided by affiliated OJP users) to access routing information (in their home language), which goes beyond the home region but covers the holistic LinkingAlps region also across national borders and incorporates multimodal services. This capability provides an unprecedented service range and quality for End-Users in the OJP-ecosystem.

1.6.2 OJP User

OJP Users in the LinkingAlps distributed journey planning system, are located on the local front-end of the LinkingAlps distributed journey planning system and enable the End-User to enquire trip plans via an End-User application. The OJP User can either be a third party, which is connected via an OJP interface to a participating system of the distributed system, or it can also be a service provider who is an integrated participant of the OJP system.

OJP Users act as clients of the home system (for local trip requests) and of the distributing system (for distributed trip requests), which are providing the route results for End-User requests.

OJP Users are interested to constantly improve the quality and quantity of their services to stay competitive and attractive for End-Users. As part of the OJP-ecosystem, OJP Users benefit on one side from a significantly extended service coverage, due to the access to an interlinked journey planning network which goes far beyond the coverage of their home journey planning system. On the other side, the quality of information for End-Users can also be enhanced, due to the provision of seamless, transnational routing information (which is a clear USP on the current journey planer market), as well as due to an elevated accuracy and actuality of provided information.

For the OJP User the requirements document is important, because it outlines the capabilities of the LinkingAlps distributed journey planning system as well as lays out the technical requirements for an integration into a local journey planning system.

Within the LinkingAlps distributed journey planning system, STA will act as OJP User as well as OJP Implementer and provide access to the LinkingAlps service over their End-User application. In future also third parties can connect over the OJP interface to a participating system and provide an End-User application.

1.6.3 OJP implementers

OJP Implementers are travel information service provider, which are implementing an OJP service exchange, which is in most cases on the back-end of an OJP User service.

An OJP Implementer provides a journey planning engine with OJP capabilities. OJP Implementers base their services on active or a passive system. In case of an active system, it integrates the routing information from several local journey planners to a combined seamless route. The active system is composed of a passive system and a distributing system, which communicates through an OJP interface. Via the distributing system of the active systems, OJP Implementers are able to detect journey requests and responses also from adjacent or remote regions and create OJP Trips.

OJP Implementers who work as passive system within the LinkingAlps distributed journey planning system, act as local journey planners (LJP) with an OJP interface (API) being able to respond to requests from distributing systems of active systems. Thus, passive OJP Implementers can be considered as information source within the OJP-ecosystem without distributed journey planning capabilities.

The LinkingAlps distributed journey planning system provides for OJP Implementers on one side the opportunity to gain experience as well as jointly innovative existing journey planning systems, to stay competitive as well as meet the predefined minimum standards defined by the Delegate Regulation 2017/1926 on the provision of EU-wide multimodal travel information. The requirements document was mainly developed for OJP Implementers, as they are the core executing actors to enable and operate the LinkingAlps distributed journey planning system. For OJP Implementers the requirements document provides a compilation of explanations and technical minimum requirements, which are needed to develop an active or passive OJP-enabled journey planning system.

Thus, for the OJP implementer the requirements are essential, since they are laying out and regulating the connection and the communication between the active and passive systems. Furthermore, the integration of the routing information has to be the same for all systems/ OJP implementers involved, which will be mapped in this document.

In the LinkingAlps consortium SBB, STA, VAO are OJP implementers as active systems. ARIA, 5T/CMTo and RRA-LUR are OJP Implementers as passive systems.

1.6.4 Supporting stakeholders

In addition to the enabling stakeholders (as outlined above), who interact and cooperate within the LinkingAlps distributed journey planning system, in order to provide an End-User service, a range of supporting stakeholders are involved in the LinkingAlps project, who mainly engage in consulting, research and development activities to drive the progress of the project.

On one side, actors from research and academia are involved to provide a status as well as contextual analysis of the novel approach of distributed journey planning with OJP. Technical research actors investigate and provide input on necessary technical configurations, such as the OJP Profile. Within the LinkingAlps project, UM-FGPA, CEREMA, LINKS and BLIC support the project with input.

Furthermore, actors from the policy and decision-making level support the LinkingAlps project with input and feedback on regulative frameworks relevant for the success of the project. In LinkingAlps the FoT, CMTo and AustriaTech are relevant partners for these concerns.

The requirements document is mainly relevant for technical research partners but holds also essential basic knowledge for the other supporting stakeholders, due to the fact, that the requirements document constitutes the “backbone” of the LinkingAlps distributed journey planning system.

1.7 Definitions

The relevant definitions for terms used in this requirements document are included in the global glossary, which is available at this stage of the project in the project internal document repository.

For a public version consult the LinkingAlps OJP Profile.

1.8 References

The requirements and specifications described in this document are based, next to the internal technical discussion and work of the LinkingAlps consortium, on some relevant standards and specifications that are thematically related.

The main references for this document are the following:

- CEN/TS 17118:2017: Public transport - Open API for distributed journey planning
- Network and Timetable Exchange (NeTEx) - Passenger Information European Profile (prTS 16614:PI Profile - TC WI 00278457)
- LinkingDanube specification: LinkingDanube Deliverable 5.1.2 (Final report on the development phase) + Annexes
- EU-SPIRIT European travel information network: OJP for EU-Spirit - OJP Profile Definition and Implementation Concept for EU-Spirit (Version 1.209/03/2020)CEN/TS 17118:2017: Public transport - Open API for distributed journey planning

The table below includes the public links to the aforementioned CEN/TS 17118:2017: Public transport - Open API for distributed journey planners and the related xsd-files. Additional technical discussion and information on the standard can be also found in the related forum of VDV and the Github.

CEN/TS 17118:2017	http://www.normes-donnees-tc.org/wp-content/uploads/2017/01/TC_278_WI_00278420_E-RS-170118-final3.pdf
related xsd-Files (1.0)	http://www.normes-donnees-tc.org/wp-content/uploads/2017/01/OJP-xsd_CEN-2016.zip
Forum discussion related to CEN/TS 17118:2017	https://forum.vdv.de/viewforum.php?f=88
GitHub XSD (1.1)	https://github.com/VDVde/OJP

Table 4: Publicly available references

2 System Overview

2.1 General considerations for the distributed journey planning

The distributed journey planning system as described and specified in this document is based on the CEN/TC 17118:2018 for "Open API for Distributed Journey Planning" (in the following referred to as "OJP Standard"). In CEN/TC 17118:2018 the Distributed Journey Planning is described as a network of journey planners, not just depending on the availability of an API for the exchange of data, but furthermore requiring a system that will process the end-user's enquiry appropriately, split the routing request and send specific requests to one or more other Local Journey Planners, followed by the merging of the responses with data from the own repositories on order to create a seamless journey plan provided to the end-user.

For the underlying technical architecture there are different approaches, which are out of the scope of the OJP standard itself. The technical architecture that LinkingAlps is following is described in chapter 2.3. System architecture overview.

One of the key considerations for building a distributed journey planning system according to the OJP standard is to define what supporting data (metadata) is required and where it is to be held. In the LinkingAlps environment only the exchange point lists and the gazetteers are defined as supporting data. Following a fully-distributed technical architecture the supporting data is maintained locally at the Local Journey Planners (see also 2.3 System architecture overview).

Figure 1 below shows the general flow of OJP messages within a distributed environment according to CEN/TC 17118:2018 for "Open API for Distributed Journey Planning". As stated before, for calculating a journey in a distributed environment the complete trip is not calculated within one single system, instead the planning task is split and distributed to several Local Journey Planner. Generally spoken, following the flow chart of the OJP standard (see figure X) the client system **(End-User application)** sends a *TripRequest* message to its enquirer's home system (**an active system**) via the **OJP Interface**. The **active system (OJP Router)** forwards this request to the **distributing system**. Within the distributing system the business logic to split the journey planning request into pieces and to find relevant passive systems (=OJP responder) has to be implemented. So the distributing system sends the request for journey-parts (aka trip legs) to the corresponding passive systems, receives the responses and is able to create OJP trip compositions. For planning a trip one, two or more OJP responders may be involved and each of them may get more than one request before the total trip is completely calculated. As a general consideration it is clear that the **distributing system** will need to know from which journey planning systems it needs to draw all relevant information and it needs to be able to communicate with the relevant responding systems (=OJP responder) using the OJP interface (API).

The implementation and requirements related to the different components in LinkingAlps is described in chapter 3.

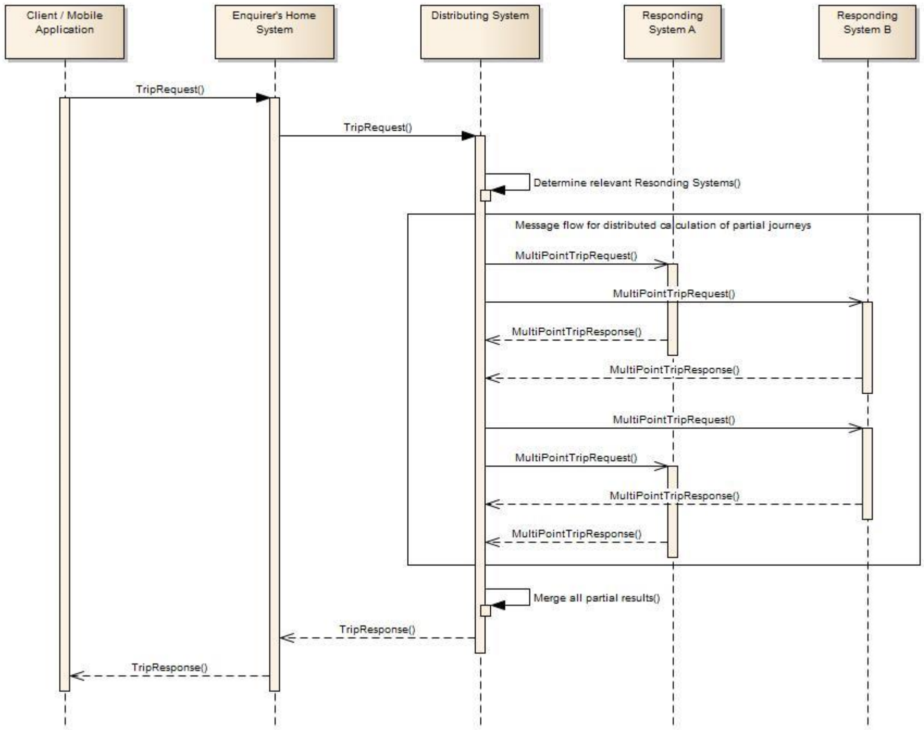


Figure 1: General flow diagram of OJP messages in distributed journey planning environments (source: CEN/TC 17118:2018 for "Open API for Distributed Journey Planning")

The OJP standard (CEN/TC 17118:2018 for "Open API for Distributed Journey Planning") distinguishes between three types of regions in relation to the geographical context of a distributed journey planning system:

- the **local region** – the territory for which the journey planner itself can plan trips without information from other systems
- the **adjacent region**s – the regions which are adjacent to the local region and have their own "local" journey planning systems
- **remote regions** – the regions which have their own "local" journey planning systems but which are not adjacent to the local region

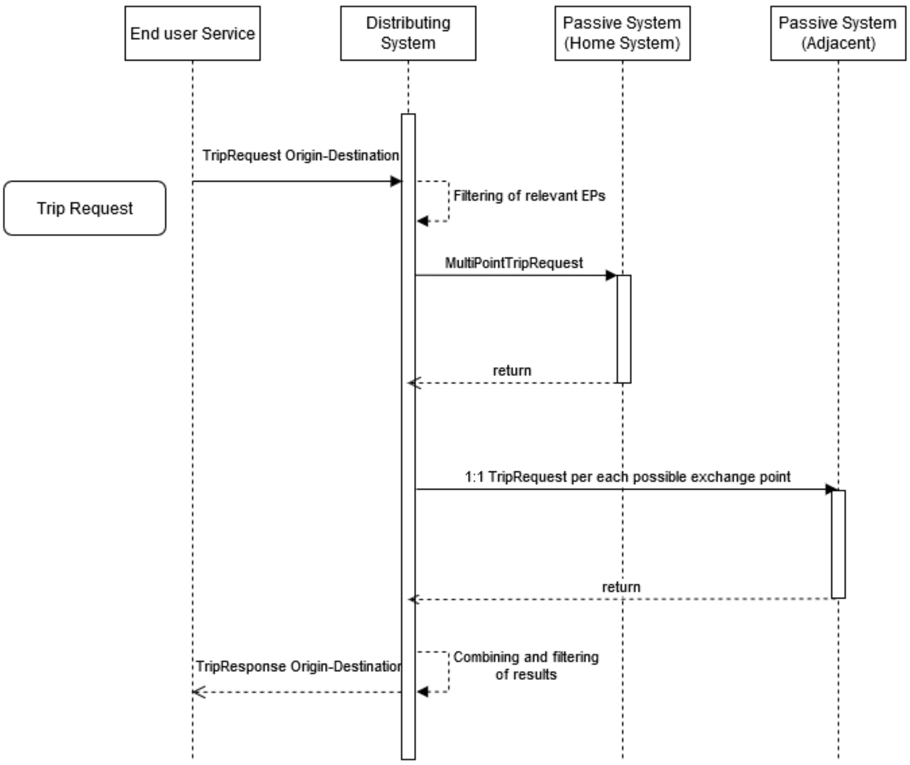
Following this the OJP standard defines five typical distributed journey planning scenarios (in addition to the non-distributed scenario of a local point to a local point), which are handling differently (depending on the underlying technical architecture defined):

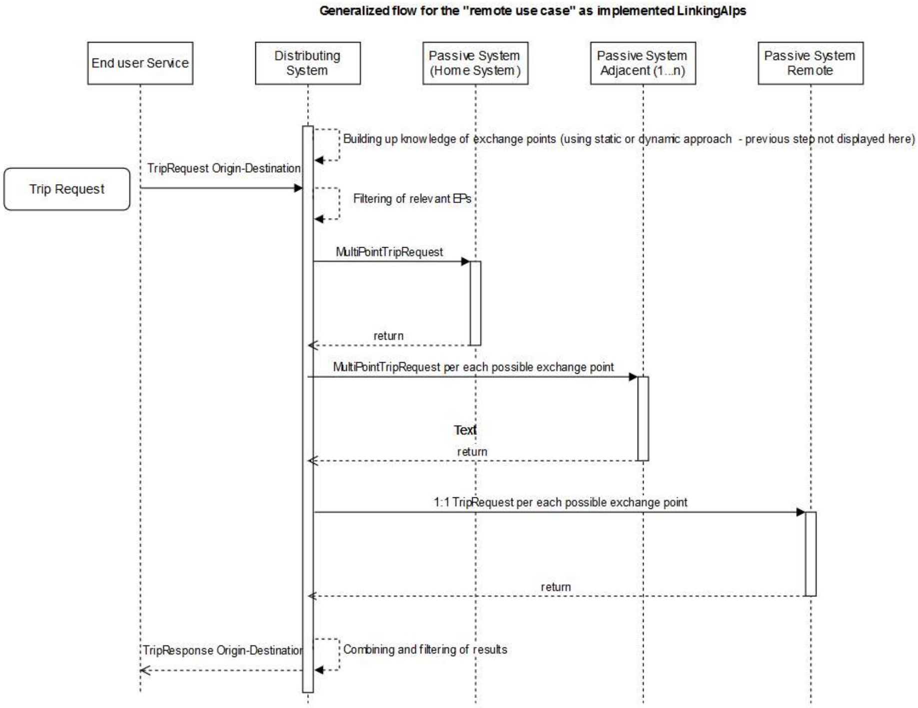
- local point to remote point.
- local point to adjacent region point.
- remote point to remote point (same region).
- remote point to remote point (different regions)
- remote point to remote point (different but adjacent regions)

So for the general request flow and the underlying technical architecture, as it is described for LinkingAlps in the following chapters, we can basically distinguish between the **"remote use case"** and the **"adjacent use case".** The distribution between these two basic use cases are described in the OJP standard is especially relevant for the definition and handling of exchange points (see chapter 3.8 and 4.1.3). For LinkingAlps some general use cases (basing on this) were described explicitly for the project (incl. in Deliverable D.T1.2.1.), which will be followed by further technical use cases in the project

The following flow charts describe in a generalized way the principle handling of the adjacent and the remote scenarios in LinkingAlps.

Generalized flow of the "adjacent use case" as implemented in LinkingAlps





2.2 System environment

The LinkingAlps Distributed Journey Planning Service (DRJP) is based on a distributed system that is currently under joint development by a total of six physically remote OJP implementers, namely SBB, VAO, STA, LUR, ARIA, CMTo/5T, managing the local JPs that will be linked together. This distributed network of OJP implementers collaborates in order to exchange travel information and routing results across the Alpine region with links between Switzerland, Austria, Italy and Slovenia. In the following sections, the participating OJP implementers are described in terms of their ingested role in the DRJP, their geographical coverage, the main sources of data and an overview of the modes provided.

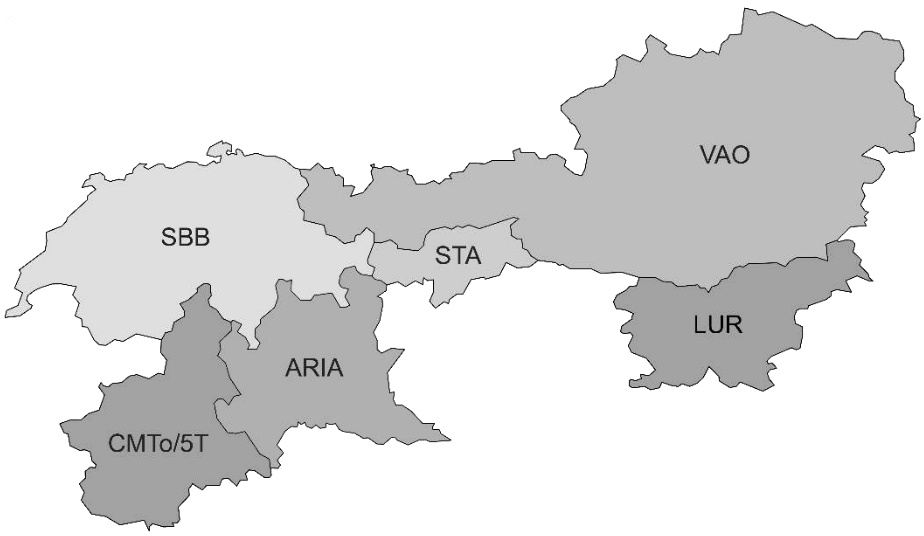


Figure 2: LinkingAlps coverage area (source: LinkingAlps, 2020)

SBB (CH)

Within the LinkingAlps project, SBB will participate as an active system. The journey planner (JP) of SBB includes the public transport supply of the Switzerland territory and bordering areas as well as international train connections to and from surrounding countries. The swiss public transport network has a total length of 24,500 kilometers and has more than 28,000 public transport station and stops. In terms of transnational coverage, the SBB JP provides connections between Switzerland and the Greater area of Lac Lemans (soon the whole border belt to France), South Germany and parts of Italy and includes International trains. Data related to Swiss transport supply are owned by the Swiss Government (OGD) and provided by every licensed transport company that must supply timetables and real-time data by law. Interconnection and transnational services from abroad are managed with pooled data from Europäisches Fahrplanzentrum (EFZ), the European timetable centre and direct data exchange between Switzerland and France, and Switzerland and Germany. For Linking Alps only data obtained directly from the partners will be used. Public transport modes and sub-modes that are provided in the SBB current application are rail, tram, metro, bus, water and cableway. Transfer modes are currently limited to walk and remain in vehicle. Nevertheless, some extensions are planned with the ongoing expansion of the OJP server.

VAO (AT)

Within the LinkingAlps project, VAO will participate as an active system. The routing service of VAO covers all of Austria and some limited information for long distance trains outside of the country. Transnational connections for a certain geographical buffer zone around Austria are provided as well. The routing graph for individual transport and for visualisation of public transport services is based on the Austrian Graph Integration Platform (GIP), the reference system of Austrian public authorities for transport infrastructure data. Other data and information come directly from the operators and VAO partners (mainly the Austrian public transport associations, the federal railways, the federal office of metrology, the surveying Austrian highway operator). The VAO system was developed involving all relevant Austrian transport infrastructure and service providers, to build up a common back-end system with the highest quality traffic and traveller information. VAO provides multimodal route information. VAO services can be used by all VAO partners as well as third parties. The system is managing almost 70,000 stops, more than 8,000 kilometers of rail, over 2,000 kilometers of motorway and expressways, around 250,000 kilometers of low-level roads, almost 1,000 parking spaces, parking garages and rest areas, 4,000 bicycle parking facilities, 3 million of addresses and over 5,000 points of interest. The VAO service supports bus, trolley bus, tram, coach, rail, intercity rail, urban rail, metro, water, cableway and funicular. The transfer modes provided are walk, park and ride, bike and ride, bike hire, protected connection, guaranteed connection, remain in vehicle, change within the vehicle, also bike sharing services that are included as transfer modes in the routing calculation managing them as a kind of bike hire.

STA (IT)

Within the LinkingAlps project, STA will participate as an active system. STA are involved within strategic transport projects on behalf of Autonomous Province of Bolzano and owns railway infrastructure and trains, plans strategic public transport projects and runs the information systems including the public transport travel planner. The STA routing service provides information on public transport multimodal solutions for the whole Bolzano Province (Trentino and Belluno are not provided for the moment), the South Tyrol and the Tyrol areas. The transport data are owned by the Province of Bolzano and provided by both the Province of Bolzano and RFI (Rete Ferroviaria Italiana, the Italian manager of rail infrastructures). It includes more than 20 operators, 3,500 stops, 6,000 stop points, 4 train lines, 200 bus lines, 5 cable ways, 1 funicular. At a transnational level some trains stopping in Bolzano and directed to Austria (Brennero line), a bus line to Switzerland and some cross-border connections to Innsbruck, Munich and Lienz are provided. Interconnections are managed by pooling data from other regions (Trentino, Belluno and Tyrol). In terms of modes, the STA service supports regional buses, urban buses, short distance trains, long distance trains (stopping in Bolzano), cableways and funicular. Transfer modes supports walk, remain in vehicle, guaranteed connection and protected connection to transfer between different modes of transport.

RRA-LUR (SLO)

The Regional Development Agency of the Ljubljana Urban Region (RRA LUR) is responsible for regional development in Slovenia. LUR was in the past operating the journey planner AtoB Ljubljana. At the moment a national journey planner is under development (AtoB Slovenia), which is provided directly by the Slovenian Ministry of Infrastructure and will cover all of Slovenia. The transport data are provided by the transport operators directly to the Ministry for Infrastructure that is the official data owner. The service provides information for bus, coach, rail, intercity rail and urban rail modes. Submodes are available for bus, coach and rail. The JP currently supports walk and cycle for both transferring and moving individually.

ARIA (IT)
Within the LinkingAlps project, ARIA will update the current JP to the common OJP interface as a passive system. ARIA is a public-capital service company as an initiative of the Lombardy Region that designs and implements ICT Systems for the Regional Government and is in charge, for example, of developing the following regional portals: Muoversi in Lombardia (the end-user interface of the online regional JP), E015 Digital Ecosystem, Lombardy Region Open data portal and the GIS Geoportal. The JP includes Lombardy public transport services in addition to some national and international trains lines, managing 83 different transport operators with 1,700 lines and 35,000 stops. The service is provided at a regional scale (Lombardy region), furthermore national and international trains lines with at least one station within the region are included. Transport data are owned and provided to ARIA by the operators in very different formats. Data provision and updates are then depending on the transport companies, that can be also very small enterprises with a scarce level of digitalisation. The JP supports six classes of modes structured in train, metro, bus, tramway, water and others (e.g. Cable car, Funicular). In terms of transfer modes, the service provides walking transfers but only between the origin of the journey and the first boarding stop of the solution, the last stop of the solution and the destination of the journey, or between intermediate stops with a pre-defined footpath, since they currently do not manage routing on a map.

CMTo/5T (IT)
Within the LinkingAlps project, CMTo/5T will contribute in the role of a passive system providing access to data via the OJP interface. CMTo is a local authority at sub-regional level and has the following purposes: metropolitan strategic planning, management of public services and infrastructures; mobility and viability planning, mobility management. CMTo in house company 5T is appointed by CMTo and Piemonte Region to develop and operate the Piemonte multimodal trip planner "Muoversi in Piemonte" managing data from 96 regional transport operators with 1,374 routes and more than 20,000 stops The coverage scale is regional and all Piemonte PT is included and only a part of Lombardy Region connections are currently included in the JP. Transport data are owned and provided by transport operators and some local authorities in GTFS format. Data quality is then sometimes affected by the scarce level of digitalisation of some transport companies, that can be also very small enterprises. The end user interface currently supports bus, tram, rail (intercity rail and urban rail) and metro. Regarding transfer modes, walk is currently the only available option.

2.3 System architecture overview

The system outlined in this document is the **Linking Alps Distributed Journey Planning Service (DRJP).** Is a network of existing local, regional or national travel information services (routing platforms) that collaborate on the basis of CEN **OJP exchange interface** (CEN/TC 17118:2018 for "Open API for Distributed Journey Planning) specified in D.T1.5.1 Specification of the API interface (including a LinkingAlps OJP Profile) in order to exchange travel information and routing results. A web-based communication network between the participating systems needs to be established as the systems are physically remote. A universal common interface for exchanging requests between the participating services needs to be specified and implemented at all participating systems.

For describing the system architecture, the following terms are used:

- **A participating system** is part of a decentralised network of journey planners (JPs) established through OJP interfaces. Participating systems can be active or passive role in the architecture, depending on their tasks. Participating services are distinguished according to the functionality and scope to active and passive systems.
- A **Local Journey Planner** is a system with a routing engine and access to multimodal data with a particular local, regional or national coverage. "Local" underlines its focus on a specific coverage that is limited. A LJP itself has no transnational (or distributed) OJP routing capability.
- **An active system** is a travel information service, in particular a journey planner, to which the end user is connected to (that means it is the enquirer's home system). It is providing an openAPI service (exchange service), the OJP interface, and is actively requesting the information from other services by using a distributing system. Hence the active system contains a distributing system that has the distribution logic in order to gather the needed information. The active system further integrates the routing information from several local journey planners (active or passive systems) to a combined (seamless) route. Doing so, it has an OJP routing algorithm facilitation the trip composition. In order to gather the required trip information in some cases the active system is also responding to requests from other systems through the OJP interface and consequently takes over the tasks of a OJP responder. In the system architecture description, it is called the OJP router indicating that it comprises an OJP interface, a distributing system and OJP routing. An active system can, but must not contain the end user service as well.
- **A passive system** is a travel information service, in particular a local journey planner, that is providing an openAPI web service (exchange service) (OJP interface) so that other clients can access information from the server. Passive systems are so called **OJP responders** and deliver responses to request over OJP interface coming from active systems. Passive systems have no distributing system and do not provide an OJP routing. Both, active and passive systems can be in the role of a responding system as the communication in the network is on a peer-to-peer basis. Therefore, the term "active and passive" systems are not used in the system architecture component description.
  Besides of active and passive systems, there are further system components, like the **end user application.** The end user application provides the GUI to the end user and is able to provide a seamless routing.
- **Processes and tools/services to exchange supporting data** for the service need to be defined and agreed between the active parts of the service in order to provide a functional distributed journey planning service.

The overview of the system architecture as included below is not the “server-centered” view, but describes the LinkingAlps ecosystem and the interconnection between the different components of the network, also from a process point of view.

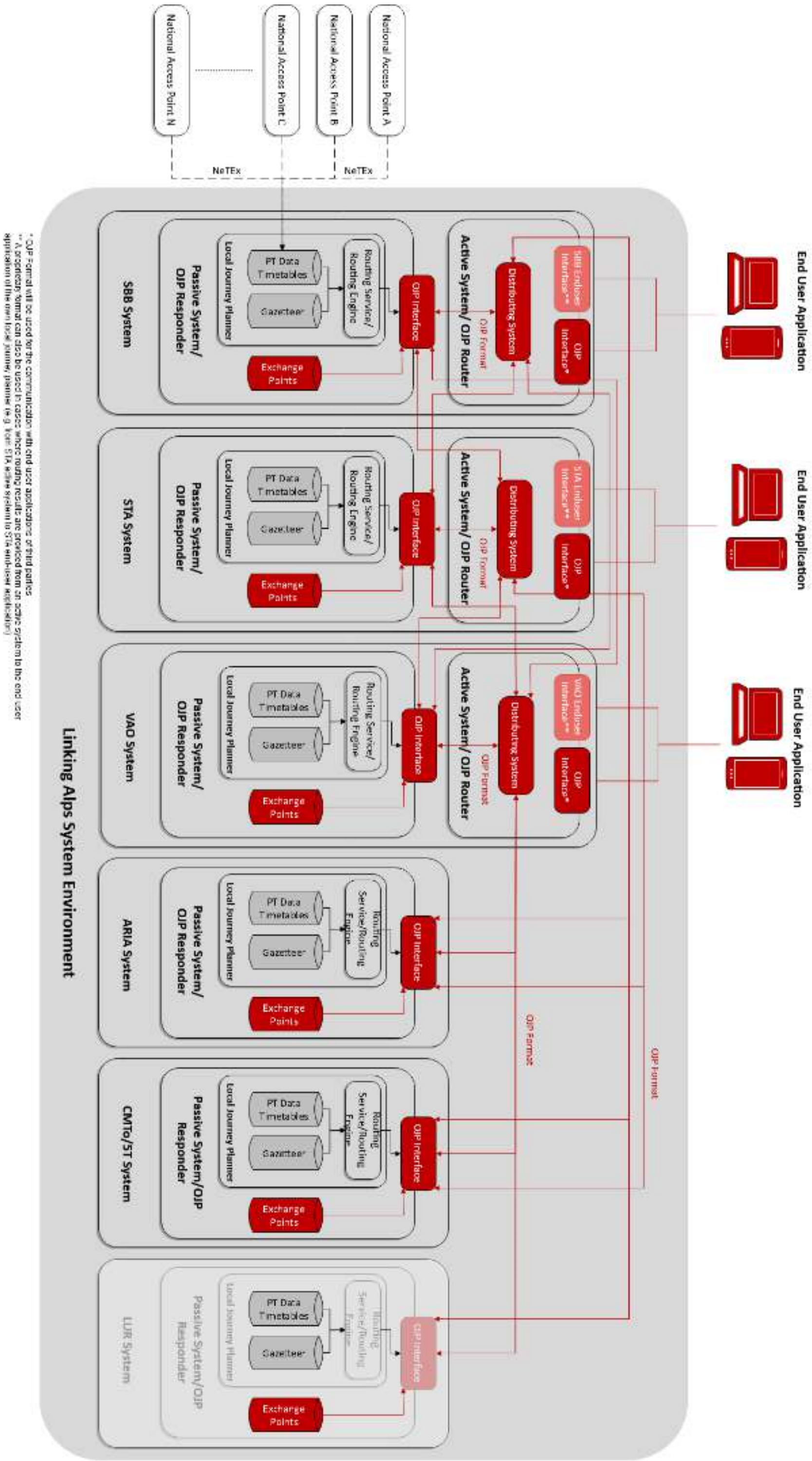


Figure 3: LinkingAlps system architecture overview

The main components of the architecture, that are also described in the following chapters are:

- End user application
- Local Journey Planner (routing service/routing engine) providing PT Data Timetables and Gazetteers
- OJP Router (active system providing the OJP routing)
- Distributing system
- OJP Responder (passive system)
- OJP interface (based on a common profile)
- OJP Format
- Exchange points
- National Access Points (supporting data)

Details for the different components together with the requirements related to them can be found in the upcoming chapters.

3 Description of the main components (incl. technical requirements)

3.1 Overview of components

In this chapter the different main components for implementing a LinkingAlps distributing journey planning system are described. Each subchapter includes the technical requirements for implementing this part, as agreed in the LinkingAlps consortium.

The components described are taken into account in the described system architecture as laid down in the previous chapter 2.3 System architecture overview.

The following chapters cover:

- End user application
- OJP Responder (alias passive system)
- OJP router (alias active system)
- Gazetteers
- Exchange point lists/service
- Distributing System
- OJP Interface (OJP API based on a LinkingAlps OJP Profile)

3.2 End user application (client)

The end user is a public person (traveller) asking for information at an end user service (application for journey planning). He or she is the actor in the use cases and the enquirer of a trip plan. He or she are typically a cross-regional or transnational commuter or tourist that wants to receive multimodal travel information beyond the known regional borders.

The end user application in concrete is the travel information application to which the end user has access to integrated information generated by the OJP services. The provider of the end user application are the so called "OJP users" in the LinkingAlps project. It is the client of the home system (for local trip requests) and of the distributing system (for distributed trip requests) that are providing the results of the distributed journey planning enquiry to the end user through a GUI. The end user service can be within the party/body of the home system or can be third party services. In the below task only the distributed journey planning case is regarded.

Req_LA_EU_018	he end user application for LocationInformationRequest should be ready to deal with a result set, that has multiple responses with the maximum probability. This is e.g. due to the tokenisation at the beginning of a search (e.g. "St. Gallen, Haggen" and "St. Gallen Haggen" are identical for the search engines. Therefore NumberOfResults=1 should not be set and also the application should be able to parse a full set of responses	Agreed	

3.3 OJP Responder (alias Passive Systems - Responding system)

Following the LinkingAlps system architecture as outlined 2.3 System architecture overview an OJP responder is a passive system within the decentralised network of journey planners established through OJP. Typically, it is the back-end of a local journey planner (LJP) (see also chapter 2.3 System architecture overview for the further definitions), meaning in concrete the server providing clients with a routing service. The OJP responder is providing an openAPI web service (exchange service), the so called OJP interface, so that other clients can access information from the server.

Within the LinkingAlps architecture the OJP responder has the role of an information source in the network and provides responses to the requests of the active systems (=OJP routers) over the OJP interface. To facilitate the communication an OJP interface (compliant with the defined requirements in this document and following the LinkingAlps OJP Profile) is implemented at each participating OJP responder.

OJP responders do not have a distributing system and do not provide an OJP routing (meaning a distributed routing, as it is provided by the active systems).

OJP responders have to provide a routing engine, access to the multimodal travel data with a particular coverage, a list of exchange points related to the particular coverage and a gazetteer, containing mainly local geolocations related to the Local Journey Planner.

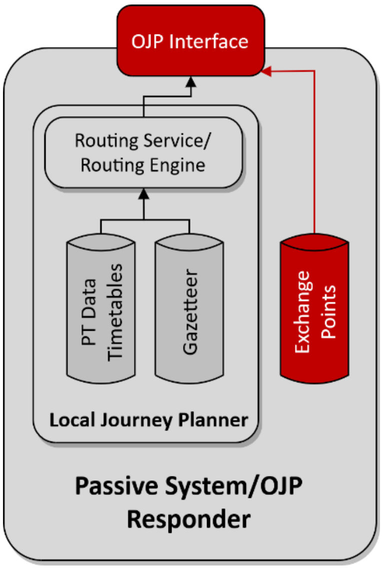


Figure 4: Passive System/OJP Responder (as part of the LinkingAlps architecture)

Preconditions:
Within the LinkingAlps distributed journey planning environment, a Local Journey Planning (LJP) that is participating as an OJP Responder (alias passive system) is considered to hold and process stops and public transport timetable data themselves to create itineraries that can be combined with others in the distributing system.

According to this, a passive system is required:

- To own a proper travel planner engine with the dataset required;
- To manage local gazetteers in its system;
- To collaborate to harmonize local gazetteers as described in chapter 3.7 System-wide gazetteers;
- To define its exchange points, according with the definitions in chapter 3.8, comparing their stops with those of the other partners;
- To manage exchange points, as required in LinkingAlps OJP profile, in its system;
- To manage global IDs, in particular for Exchange points.

Next to the general requirements as defined in this document for all participating systems the passive systems have to fulfil some **additional minimum requirements to enable the distributing routing based on OJP**.

This is mainly linked to the provision of exchange points:

- In LinkingAlps the trunk legs necessary to provide the distributed journey planning are held with the passive systems and provided upon request to the active systems.

- Within LinkingAlps two pre-defined approaches exist to gather exchange points needed for the route calculation: the dynamic and the static approach (see also chapters 4.1.2 Handling of long-distance transport (to calculate the "trunk" leg) & exchange points for trunk leg calculation and 4.1.3Handling of exchange points & relevant requests) for description of both approaches). **Passive system must be able to support both approaches, leaving the choice of the approach to the single active system**.
    - **For the dynamic approach, a request including a place reference (origin or destination) is sent to the passive system**. In this case, the passive systems have to provide a subset of exchange points based on the request received by the active system and it have to return an estimated travel time between each exchange point and the origin/destination (TravelDurationEstimates parameter). The passive system has to send back a short list of exchange points selected by a criteria depending on the received place reference. Depending on the criteria (e.g. fastest trip) a prior calculation of the trip duration from each possible stop (or other place) to each exchange point might be necessary. Considering that filters applied by end users can introduce different criteria from the fastest trip, the matrix "exchange point"/"place reference" would grow in dimensions. The heuristic solution identified in LinkingAlps project consists in three steps:
        - Step 1: A filter of exchange points has to be performed considering the filter(s) applied by end user (e.g. no trains) and communicated by the active system.
        - Step 2: An algorithm based on the distance between the exchange point and the place reference is applied (only one criteria is considered). A list of top scored exchange points is calculated and provided.
        - Step 3: A list of, few, relevant exchange points has to be provided in addition even if the score is slow.
    - **For the static approach, a request without a place reference (origin or destination) is sent to the passive system**. In this case, the passive systems have to provide all the exchange points based on the request received by the active system. With the static approach the estimated travel time (TravelDurationEstimates) is not provided in the exchange point response of passive systems, because the calculation of the travel duration takes place on active system level. The active system (in concrete the distributing system part of the active system) chooses among the list of exchange points provided.
- The filtering of the exchange points according to the defined algorithms is in the responsibility of the active systems, while passive systems should not apply other filters than those present in the request.

Other requirements specifically referring to passive systems and covered by other chapters of this specification document are summarized in the Annex.

3.4 OJP router (alias Active System)

An OJP router is the active system in the decentralised network of journey planners. It is the component that is providing an OJP service (distributed journey planning results) to an end user application. It consists of the following major components:

- OJP responder/ passive system
- Distributing system

In other words, the OJP responder equals a passive system including an OJP interface to communicate with the distributing system. The distributing system is also part of the OJP responder. Hence, the tasks and functionality of an active system is enlarged through the tasks and functionality of the distributing system.

Tasks:

- The active system (AS) provides an OJP interface (following the OJP Profile specification) to communicate with the end user application (i.e. to receive the journey planning details and to pass back the results of the distributed journey planning to the end user application)
- The active system (AS) passes the user enquiry details to the distributing system for distributed trip calculation by using the OJP interface as well.
- The active system (AS) provides the journey planning results to the end user application via the OJP interface
- The active system (AS) is using key-based authentication within the OJP interface
- The active systems (AS) provides a possibility to cover the long-distance use case. This requires different handling of trunk legs within the active systems, i.e. the distributing system that is described under chapter 2.3.2.2. Handling of long-distance transport (to calculate the "trunk" leg) & exchange points.
- The actives system (AS) contains or is connected to a distributing system, that is a the core for the distributed trip calculation.
- The active system (AS) provides the same requirements as a passive system and takes over the tasks when requested by other active systems.
- From a client/server perspective, the OJP router (active system) is the server providing clients (end user applications) with OJP routing service.

Precondition:
Organisational: The active system relies on the passive systems in the network for the distributed trip calculation. Therefore SLAs should be conclude in order to guarantee service levels and quality to end users.

Req_LA_AS_007	The active system should deal with the possibility, to ignore the NumberOfResults. The demonstrator case is LocationInformationRequest with "St. Gallen, Haggen", "St. Gallen Haggen". These two stops exist and will have maximum probability. Therefore they must be sent back.	Agreed	

3.5 Distributing system

The distributing system is part of the OJP router (active system). It includes the logic of splitting up the overall trip request and handing sub trip requests over to relevant passive systems in the network of OJP responders in order to enable distributed route calculation. Several distributing systems can exist in the network (de-central architecture).

The distributing system exchanges requests via the OJP interface by applying the OJP profile. It uses the OJP interface for

- Providing overall trip results to requesting systems
- Communication with all passive systems in the network for the reason of
    - trip calculation
    - Identifying relevant EPs (dynamic approach) or creating overall EP-knowledge (static approach)
    - Identifying location points and building up a gazetteer
    - requesting departure/arrival boards

The distributing system is authorised to call the passive system's endpoint through API keys, which have to be managed by the distributing system in its requests.

The distributing system is able to handle one of the defined approaches of trunk legs calculation and handling of exchange points described under chapter 4.1.2 Handling of long-distance transport (to calculate the "trunk" leg) & exchange points for trunk leg calculation. The distributing system must not handle both approaches, but one of them. Therefore the distributing systems in the network are allowed to be individual in this respect of exchange point handling and trunk leg calculation. However the passive systems need to support both approaches and the OJP profile must enable both approaches.

Tasks of the Distributing system:

- to distinguish if an user enquiry can be answered by the (home) passive system or if the start and/or the end point is located outside of the coverage are of the (home) passive system
- in a generic way to identify relevant passive systems for a certain request
- to request trip leg information from the (home) passive system
- in a generic way to request trip legs from all relevant passive systems for a certain overall route calculation
- to handle the adjacent use case and remote use case
- to use the gazetteers to define the end points of the trip
- to request from the determined responding systems to select the places corresponding to the selected origin or destination place of the user (and also to fill the gazetteers).)
- to identify the necessy passive systems based on gazetteer information and general connectivity between the regions of the passive system.
- to collect and/request exchange points from passive systems to identify the relevant passive systems to answer the enquiry
- to collect exchange points (regularly) to bulid up knowledge of exchange points of all passive systems (request ALL exchange points)
- to integrate the exchange points in the stop database and calculate the trunk leg within the (home) router OR to request a specific set of exchange points (run-time) based on a specific logic/search strategy
- to provide a logic/search strategy how to select the appropriate/suitable exchange points from the whole set of candidates exchange points
- based on this logic/search strategy to filter relevant/suitable exchange points the DS is calculating a fist estimate (meta-graph) for the trunk leg (refer to the different approaches for trunk leg calculation)
- to provide a logic/strategy to optimise the trip requests, meaning to identify the most suitable exchange points (e.g. by using the TravelDurationEstiamte and potentially other attributes in the exchange point list)
- to orchestrate the requests according to the implemented logic/search strategy that is compliant with the OJP profile
- the distributing system exchanges the message in a sequence that is defined in a commonly supported request flow
- to implement the general search strategy (e.g. request with a multi-point request the passive systems one by one)
- to collect back the responses (trip legs) from OJP responders (passive systems) and integrate the results to a seamless trip plan
- to sort and priories and select trips from the collected results
- to send the combined trips as result to the active system via the OJP interface
- the DS is able to glue trips together.

Precondition:

- The distributing system can gain knowledge of all exchange points between relevant journey planning systems (passive systems) and its relying on an up-to-date exchange point service or repository of the passive systems. → Req_LA_EPD_001
- The OJP profile reflects both approaches for exchange point gathering and trunk leg calculation by taking account of the needed attributes in the exchange point schema → Req_LA_LDT_001 and Req_LA_LDT_002
- An exchange point shall be defined either as stop or station with direct connection in a region/country covered by the neighbouring system → Req_LA_EPD_015
- An exchange point shall provide the possibility of changing the transport option → Req_LA_EPD_016
- All required requests are reflected in the OJP Profile
- The distributing system has access to complete gazetteers.
- The data in the passive system is either disjunct or it is clear for the DS how to cut&paste journeys together. This means the journey must be identifiable by the DS from each passive system, when it is part of the trip.

3.6 OJP Interface (OJP API based on a LinkingAlps OJP Profile)

Transit users travelling across borders often face the problem that travel information for the entire route is not visible immediately. In most cases, travellers must switch between the information systems of different operators, regions or countries in order to plan their entire journey. The LinkingAlps project addresses this problem in the Alpine Space (AS). Its aim is to create a standardised exchange service of travel information between the individual travel information service providers. In this way, information can be exchanged between different systems and compiled into a continuous travel chain. Travelers can thus view the entire trip from origin to destination on a single service. This document describes how communication and supported services are handles within LinkingAlps.

Within the context of LinkingAlps this exchange of travel information is based on the OJP (Open Journey Planer) standard version with a special profile for the supported services, fields and parameters. It therefore describes a concrete form for the usage of the OJP standard. The LinkingAlps OJP Profile aims to define a specific subset of (XML) data elements following a clearly arranged structure in accordance with the OJP standard (CEN/TS 17118:2017) and defined using XML schemas. The schemas include all functionalities required for an OJP interface in order to enable communication with the LinkingAlps distributed journey planning system. In this sense the LinkingAlps OJP Profile defines the content and to a degree the structure of the information content. The actual flow of information as well as the physical exchange format are defined in the OJP standard.

There are currently seven different OJP services described in the OJP Standard. Within the context of the LinkingAlps project six of them are initially supported, excluding the OJPFare service. However, it should be noted that this service is likely to be included at some point in the future. For the sake of complete documentation, the names of the related OJP schema files and a short description for each service are given as well.

Service name	Service in OJP CEN/TS 17118:2017	XML Schema file	Supported
OJPLocationInformation	Location information	OJP_Locations.xsd	Yes
OJPTrip	Trip request	OJP_Trips.xsd	Yes
OJPStopEvent	Departure board	OJP_StopEvents.xsd	Yes
OJPTripInfo	Trip/Vehicle information	OJP_TripInfo.xsd	Yes
OJPExchangePoints	Exchange points	OJP_Locations.xsd	Yes
OJPMultiPointTrip	Distributed journey planning	OJP_Trips.xsd	Yes
OJPFare	Ticket price calculation	OJP_Fare.xsd	No

Table 5: Overview OJP services supported in LinkingAlps

Between the OJP router, distributing system and OJP responder as well as end user application and the communication is done via a API based on a common LinkingAlps profile. This API is a standardised interface allowing communication between all distributed system parts. The interface is based on "Open API for distributed journey planning "developed by Technical Committee CEN /TC 278 (2017). The LinkingAlps OJP Profile is compliant with the mentioned TS (technical specification). The OJP service shall be implemented at OJP routers (active systems), distributing systems, OJP responders (passive systems) according to an agreed and aligned LinkingAlps OJP Profile.

A detailed specification to the LinkingAlps OJP Profile can be found in the Deliverable DT1.5.1 and related documents.

3.7 System-wide gazetteers

Definitions

Gazetteers are important tools used in a wide variety of workflows that depend on linking natural language text to geographical space and its type information [Elise Acheson, Stefano De Sabbata, Ross S. Purves, A quantitative analysis of global gazetteers: Patterns of coverage for common feature types, Computers, Environment and Urban Systems, 2017, https://doi.org/10.1016/j.compenvurbsys.2017.03.007].

A gazetteer is a directory of common objects across the local journey planner systems and its system borders. It enables the active system to find the passive system for all geolocations (stops, stations, points of interest (POIs), addresses, etc.). The gazetteer acts system-wide [Glossar_V04-00]. Location identification (unique identifiers, language translations, coordinate systems & geo-locations, modelling of areas in a point representation) shall be harmonised across the systems so that it can be looked up by all distributing services.

It is a repository, held decentral within the Journey Planners or in a central database, enabling an identification of all geolocations (Stop point, Stop place, Topographic Place, POIs, Address) across the local journey planner systems and its system borders, needed to determine the relevant journey planners in which the origin and destination are managed.

External sources, such as Global Spatial Databases (e.g. OpenStreetMap, GeoNames, Getty Thesaurus of Geographic Names) can be sources of information for the gazetteers (e.g. provision of addresses, POIs). These databases use a bottom-up methodology to collect data from multiple sources integrating it.

Coverage, availability, completeness and precision are some of the characteristics that should be analysed when the base of gazetteer information is chosen.

3.7.1 Gazetteer in the Linking Alps Distributed Journey Planning Service

The gazetteer is contained in the Local Journey Planner (Passive system/OJP Responder) of active and passive systems, as can be seen in Figure 1. The passive system/OJP responder serves as an information source, including information on exchange points, PT Data timetables and the gazetteer. The implementation of the gazetteer repository is responsibility of each active and passive system.

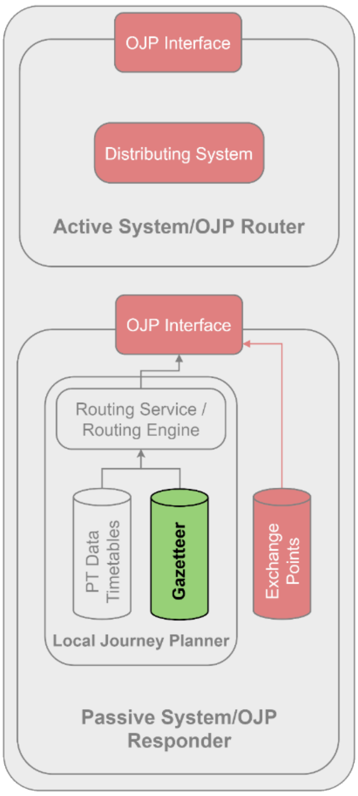


Figure 5: Gazetteer contained in the OJP Responder of an Active System

Method of handling the gazetteer

Participating systems that only function as Passive systems contain solely local geolocations on their gazetteers. Therefore, End-user application sends a Location Information Request (LIR) to a known Active system if the desired location is outside of the local gazetteer scope. Participating systems that also perform as Active Systems cache gazetteers from all other participating systems, to maintain lower response time to the request. The request structure is described in the following tables.

Table 6: Description of *LocationInformationRequestStructure,* adapted from CEN/TC 278 2017 Public transport — Open API for distributed journey planning.

LocationInformationRequestStructure				*+Structure*	Container for location information request details.
	a	*InitialInput*	-1:1	*+InitialLocationInput*	Input data for an initial location information request (see Table "Description of *InitialLocationInputStructure*").
	b	*PlaceRef*		*+PlaceRef*	Reference of a Place for which more details are to be retrieved. If Places are organised hierarchically it may be reasonable to identify the Place in a top-down approach with several steps of refining a Place on each level of hierarchy. Following this approach an initial request retrieves a first list of top-level Places (eg: streets) which are to be refined in a subsequent request to the next level (eg: house number intervals). The objects of the current level are presented to the user for selection. The object reference of the selected object is then sent in the next request for further refinement. (Reference to a stop point, stop place, geographic position, topographic place, Point of Interest, address or public name of the location).
	Restrictions		0:1	*+PlaceParam*	More parameters for restricting the request (see Table "Description of *PlaceParamStructure*").
	Extension		0:1	*xs:anyType*	Extensions.

Table 7: Description of *InitialLocationInputStructure*, adapted from CEN/TC 278 2017 Public transport — Open API for distributed journey planning

InitialLocationInputStructure			*+Structure*	Contains the parameters for the initial location request.
	LocationName	0:1	*xs:string*	Name of the location object which is looked for. This is usually the user's input. If not given, the name for the resulting location objects is not relevant.
	GeoPosition	0:1	*+GeoPosition*	Coordinate where to look for locations. If given, the result should prefer location objects near to this geoposition. (Geographical position expressed as WGS84).
	GeoRestriction	0:1	*+GeoRestrictions*	Parameters for geographical filtering restricting the search (The filter is defined by a circle, a rectangle or by a polygon).

Table 8: Description of *PlaceParamStructure*, adapted from CEN/TC 278 2017 Public transport — Open API for distributed journey planning.

PlaceParamStructure			+Structure	Contains the parameters controlling the search for location objects.
LocationDataFilter	Type	0:*	stop | address | poi | coord | TopographicPlace	Allowed location object types. If none is given, all types are allowed.
	Usage	0:1	origin | destination | via	Defines, whether location objects for origin, via, or destination are searched.
	PtMode	0:1	+PtModeFilter	Allowed public transport modes. Defines, which public transport modes have to be available at the returned location objects. Applies only to stops.
	OperatorFilter	0:1	+OperatorFilter	Filter for locations that are operated by certain organisations (An organisation code is used to uniquely reference transport companies and transport authorities. In order to have these codes remain unique across multiple systems, it is recommended that comprehensive databases (optimally national ones) of transport companies is built).
	TopographicalPlaceRef	0:*	→TopographicPlaceCode	If at least one is set, only location objects within the given localities are allowed (Identifier of a Scheduled Stop Point, Stop Place, Topographic Place, Point of Interest or Address).
	PointOfInterestFilter	0:1	+PointOfInterestFilter	Filter to narrow down POI searches (See Table "Description of PointOfInterestFilterStructure").
LocationPolicy	Language	0:1	xs:language	Preferred language in which to return text values.
	NumberOfResults	0:1	xs:positiveInteger	Maximum number of results to be returned. The service is allowed to return fewer objects if reasonable or otherwise appropriate. If the number of matching objects is expected to be large (eg: in the case that all objects should be delivered) this parameter can be used to partition the response delivery into smaller chunks. The location information service is expected to support a response volume of at least 500 location objects within one single response.
	ContinueAt	0:1	xs:nonNegativeInteger	Tells the server to skip the mentioned number of results in its response. Can be used in a follow-up request to get further results. The value is usually taken from the previous response (element *ContinueAt*, see Table "Description of PlaceInformationResponseStructure"). To request the remaining objects the request is repeated with *ContinueAt* set to the value given by the last response.
	IncludePtModes	0:1	xs:Boolean	Tells the service to include the available public transport modes at returned stops. Default is *false*.

Table 9: Description of *PointOfInterestFilterStructure*, adapted from CEN/TC 278 2017 Public transport — Open API for distributed journey planning

PointOfInterestFilterStructure			*+Structure*	Structure for filtering by point-of-interest categories.
	Exclude	0:1	*xs:boolean*	Specifies whether points of interest should be searched only within the given categories (Exclude=*false*) or do not meet any of the given categories (Exclude=true). Default is *false*.
	PointOfInterestCategory	1:*	*+PointOfInterestCategory*	List of point-of-interest categories. If the list contains more than one category the search for points of interest will combine them by a logical "OR"" in the case of Exclude=false and by a logical "AND" in the case of Exclude=true.

Table 10: Description of *PlaceInformationResponseStructure*, adapted from CEN/TC 278 2017 Public transport — Open API for distributed journey planning

PlaceInformationResponseStructure			*+Structure*	Container for the results of a location information query.
	ErrorMessage	0:*	*+ErrorMessage*	Error messages related to the processing of the entire request. Possible errors: No results to match the input data, unsupported location types, unsupported combination of input data (text string, coordinates, geographical restrictions), location of object could not be refined, usage type has been ignored, unsupported restriction by transport modes, unsupported any restrictions by localities.
	ContinueAt	0:1	*xs:nonNegativeInteger*	If this value is set the service indicates that there are more location objects to be returned than one response can transmit (due to size limits). The value of *ContinueAt* can be used in a follow-up request to get further results (see Table "Description of *PlaceParamStructure*"). It tells the server to skip the given number of results in its response.
	Place	0:*	*+PlaceResult*	The location object results found by the service. They have to be sorted by the ranking how well they match to the input data in descending order. The first result in the list matches best (see Table "Description of *PlaceResultStructure*").

Table 11: Description of *PlaceResultStructure*, adapted from CEN/TC 278 2017 Public transport — Open API for distributed journey planning.

PlaceResultStructure			*+Structure*	Result structure for a location object.
	Place	1:1	*+Place*	The Place object (re-usable base types and structures for referencing and describing locations (stop places, stops points, localities, addresses and points of interest).
	Complete	1:1	*xs:boolean*	States whether the included Place is complete or needs further refinement. Only complete Places are fully resolved and can be used in e.g. trip requests. Incomplete Places have to be refined entering them once again into a LocationInformationRequest.
	Probability	0:1	*xs:float*	Probability, that this result is the one meant by the user's input. Value should be between 0 and 1.
	Mode	0:*	*+Mode*	List of transport modes that call at this location object. This list should only be created in case of stop points or stop places – and only when explicitly requested.

Concept of the gazetteer cache

The concept of caching gazetteer data from participating (passive) systems is introduced to reduce the response time for the Location Information Request. Gazetteer cache data is exported from the passive systems. The starting points and destinations of the trip entered by the end user must be converted into unique addresses throughout the system. The address data is generally only used for pattern matching. The active system converts the address into coordinates (WGS84 format) and requests only one coordinate from the passive system at a time. Most of the systems involved already provide address data on an OSM basis, and all passive systems in LinkingAlps can also support this. A complete export of the OSM address data of all passive systems is not necessary. Instead, it is advisable that the active systems themselves retrieve the OSM data for the entire LinkingAlps area and access it locally. On the part of the passive systems, a complete export is only mandatory for stops and POIs, but not for addresses.
Alternatively, a dynamic look up of location matches from the responding systems must be implemented. The number of queries to responding system should generally be limited so that the performance is not decreased.

Figure 6 shows a concept for the Gazetteer cache mechanism. The mechanism contains methods for collection of addresses from the global spatial database (OSM) and replication of location objects from the passive systems. Replication is a repetitive maintenance task (during night and not during day) where locations (stations, stops, POIs) are exported from the passive systems and copied (inserted, updated, deleted) into all active systems implementing the Gazetteer cache.

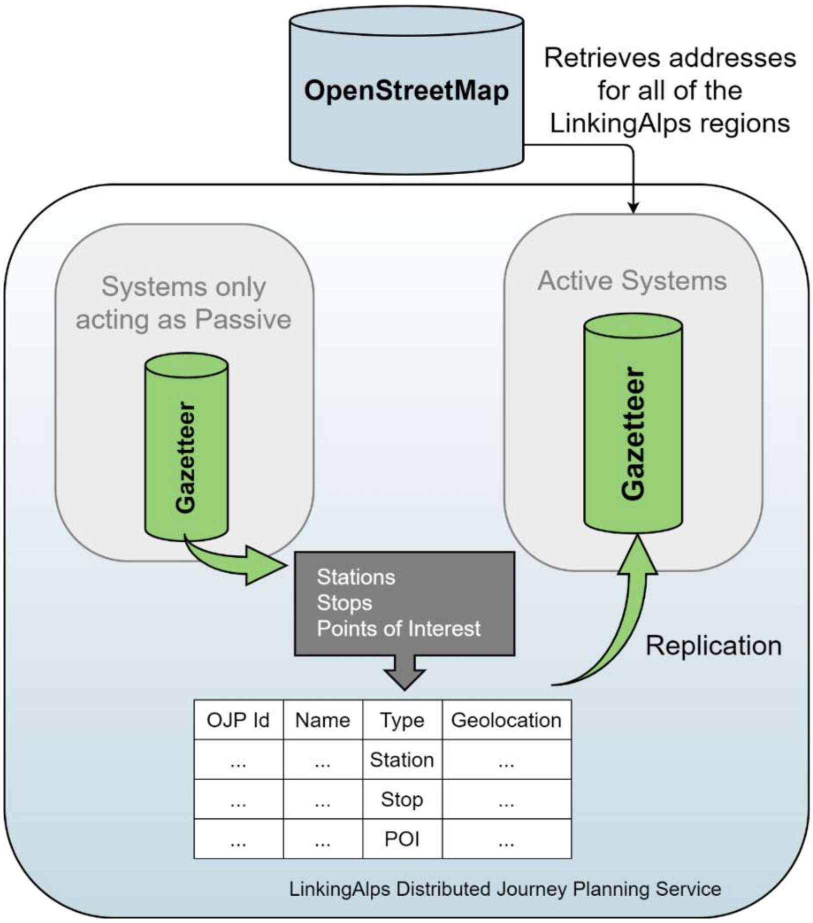


Figure 6: Conceptual overview of the gazetteer cache mechanism.

Data availability for gazetteers provided by the participating systems

The following table summarises the capability of the LinkingAlps participating systems to provide data related to geolocations (stations, stops, topographic places, addresses and Points of Interest). The external sources or databases providing the data for the addresses are indicated. In addition, the OJP implementers define which coordinate system is used in their systems for geographical locations. Address data will be based by all partners as OSM data. For VAO and ARIA it will be migrated to the gazetteers.

Table 12: Gazetteers data provided by the participant systems.

Partner/Participating system	Stations	Stops	Topographic Places	Points of Interest	Addresses	Database for addresses	Coordinate system	Comment
ARIA	✅	✅	✅	✅	⚠️ (see comment)		WGS84	*Addresses: We have legal restrictions (third party service used) with our current addresses, **but we can migrate our gazetteer to OpenStreetMap**.*
CMTo/5T	✅ (derived from GTFS* data)	✅ (derived from GTFS* data)	✅ (retrieved from OSM**)	✅ (retrieved from OSM**)	✅ (retrieved from OSM**)	OSM**, OpenAddresses	WGS84	
SBB	✅	✅	✅	⚠️	⚠️		WGS84 MRCV (from OSM**)	
STA	✅	✅	⚠️ (see comment)	✅ (Open Data Hub)	✅ (OSM**)	OSM**	WGS84 can be used	*Topographic Places: No coordinates available, but searching for a certain municipality is possible.*
VAO	✅	✅	⚠️ (see comment)	✅	⚠️ (see comment)	BEV*** (Federal Office data)	WGS84 can be used	*Topographic Places: Only key coordinates for municipalities/local communities.* *Addresses: The official address data used can't be shared, **but OpenStreetMap could be used for address data** in the gazetteers.*
LUR	LUR's Journey Planner project is yet to be launched.							
								✅ Data can be shared ⚠️ Restrictions exist / data can not be shared.

*General Transit Feed Specification.

** OpenStreetMap.

*** Bundesamt für Eich- und Vermessungswesen (BEV), Austrian Federal Office of Metrology and Surveying.

3.8 Exchange Point Database or Service

Definition of exchange points

Exchange points are "transition nodes" (Figure 7: The trip leg - trunk leg - trip leg of a journey, adapted from CEN/TS 17118:2017 Public transport — Open API for distributed journey planning.) between each of the various journey planning systems (participating systems). They can also be considered as boundary points where the trip calculation is handed over to the next journey planning system [ref. CEN/TS 17118:2017 Public transport — Open API for distributed journey planning]. Exchange points in LinkingAlps are defined as stops/stations with direct connections in the neighbouring regions/countries and the possibility to change the transport option (mode, vehicle). Generally, exchange points are defined as stops or stations where the trip leg of one participating system is connected to the trunk leg of another participating system, meaning that the line is not split. Trip leg is the local part of a trip which is calculated by a single Local Journey planning system and trunk leg is a sub-set/part of a trip-leg using a specific mode of transport [ref. Glossar_V04-00].

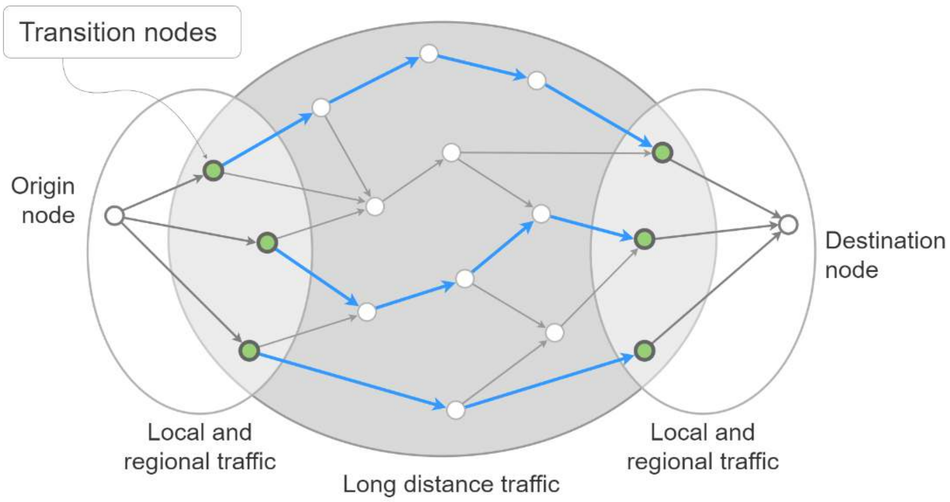


Figure 7: The trip leg - trunk leg - trip leg of a journey, adapted from CEN/TS 17118:2017 Public transport — Open API for distributed journey planning.

Exchange Points can theoretically be divided in:

- Trunk Exchange Points, being the points in each region where long-distance passengers are likely to change from trunk (long-distance) services (train or coach) from or to local services (which could include all forms of public transport, including trains and coaches);
- Adjacent Region Exchange Points being points along regional boundaries which were used to deal with trips that, whilst crossing into another region, were so "local" that they did not need to use a trunk public transport leg.

This includes regional stops which match with stops for long distance or regional stops from adjacent regions. Exchange points are mainly but not exclusively located at borders and in bigger cities [ref. Glossar_V04-00]. A defined exchange point shall exist in at least two neighbouring systems. The identification of exchange points is done by looking for stops and stations used by multiple service providers (a.k.a. areas). From the view of a local journey planer, exchange points may be inside of the covered area, outside of the covered area as well as at the border of the covered area. It is worth noting that exchange points do not only exist between services of geographical adjacent areas, but also between services of geographical remote areas in the Linking Alps Distributed Journey Planning Service (DRJP). Figure 8 presents a theoretical example for a journey from Vienna to Turin with exchange points from all the participating systems. Two sets of exchange points (VAO-ARIA-CMTo/5T and VAO-SBB-CMTo/5T) are selected as relevant (optimal) input for the journey planning.

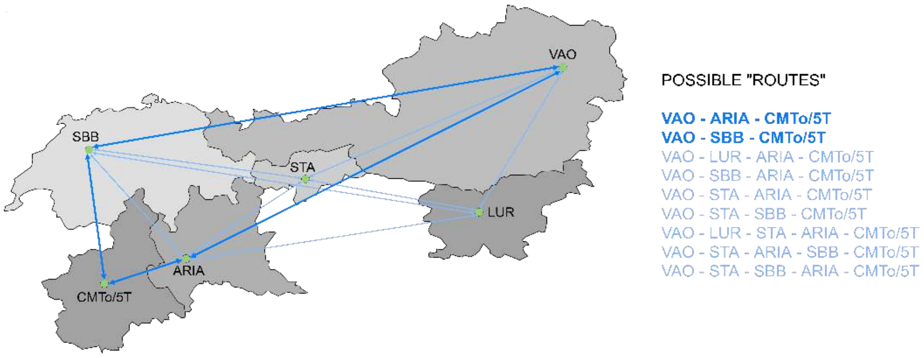


Figure 8: Theoretical example of common exchange points in the draft geographical regions shared between OJP systems in the DRJP, adapted from a meeting (source VAO).

Data structure for exchange points

Each exchange point uses global identifier (Global ID) in the LinkingAlps system. While the OJP standard document does not specify an ID structure for exchange points, an ID structure, based on the *NeTEx Passenger Information EU Profile*, shall be applied and structured as follows:

[country code]:[local code]:[object type]:[technical-identifier]:[ID provider for shared IDs]

Where:

- ***[country-code]*** is the ISO 3166-1 code (2 characters) of the country.
- ***[local-code]*** is a code uniquely identifying the locality or the provider within the country (may be a region code like the European NUTS code, an authority code, etc.). The European NUTS code is recommended here. However, this code is not mandatory if the other elements make the code unique (but a placeholder colon ":" shall still be present).

- ***[object type]*** field is optional, the recommendation is to follow the NeTEx object type (ServiceJourney, PassengerStopAssignment Line, etc. using exactly the tag format, UpperCamelCase and no space) corresponding to the XML tag, and is provided to avoid any collision of single identifiers being used for different types of objects. A small exception is defined for StopPlace in order to differentiate between monomodal and General STOP PLACEs, the [object will be MonomodalStopPlace or GeneralStopPlace instead of StopPlace].
- ***[technical-identifier]*** is a technical identifier for the object, it can be whatever code the system define (built of upper case or lower case non accented characters, numbers "-" and "_") but shall be unique for the object and durable (a single object can't change its identifier).
- ***[ID-provider-for-shared-IDs]*** is used to provide a reference to the ID provider (reference data system; e.g. a national system that provides and manages IDs) when there is one. When this non-mandatory field is present, it ensures the uniqueness and stability of the ***[technical-identifier]***.
- All the "***:***" separating characters are mandatory (even if a field is empty) and are contiguous without spaces.

Exchange Points Database or Service

An exchange points database or service can be defined as a repository or a service that is able to list the relevant exchange points of the distributed service. It can be a static system-wide database or be generated dynamically with requests for exchange points to the responding services. Exchange points are stations where the trip leg of one system is connected to the trip leg of another system. Object identifiers have to be harmonised system-wide (or at least mapped). Chapter 4.1.3 Handling of exchange points & relevant requests describes the basis for creating this database/service. The requests/responses of the LA OJP profile must be supported.

4 Interactions

The following chapters covers the technical processes that are relevant for the implementation and operation of a distributed journey planning system.

It describes the modalities how information is exchanged, what are preconditions and how the handling is done.

In concrete this covers:

- The interaction between the OJP router (alias active system) and the OJP responder (alias passive system)
    - The exchange of messages using the OJP interface
    - The handling and calculation of long-distance transport connections
    - The handling of Exchange Points
    - The orchestration (including the supported request flows and their sequence)
- The interaction between the OJP router (alias active system) and end user applications
- The authorisation, authentication and compression of the network
- Aspects related to Service and Data quality

4.1 Interaction of OJP router and OJP responder

Looking at the LinkingAlps system environment this interaction between OJP router and OJP responder (or between active and passive systems) cover the exchange of messages via the OJP interface (according to a specific defined sequence, following the defined OJP profile). In concrete this part covers the distributed route calculation, which has specific aspects like specific approaches for the handling and retrieving of exchange points or the different methods for the calculation and handling of long-distance transport.

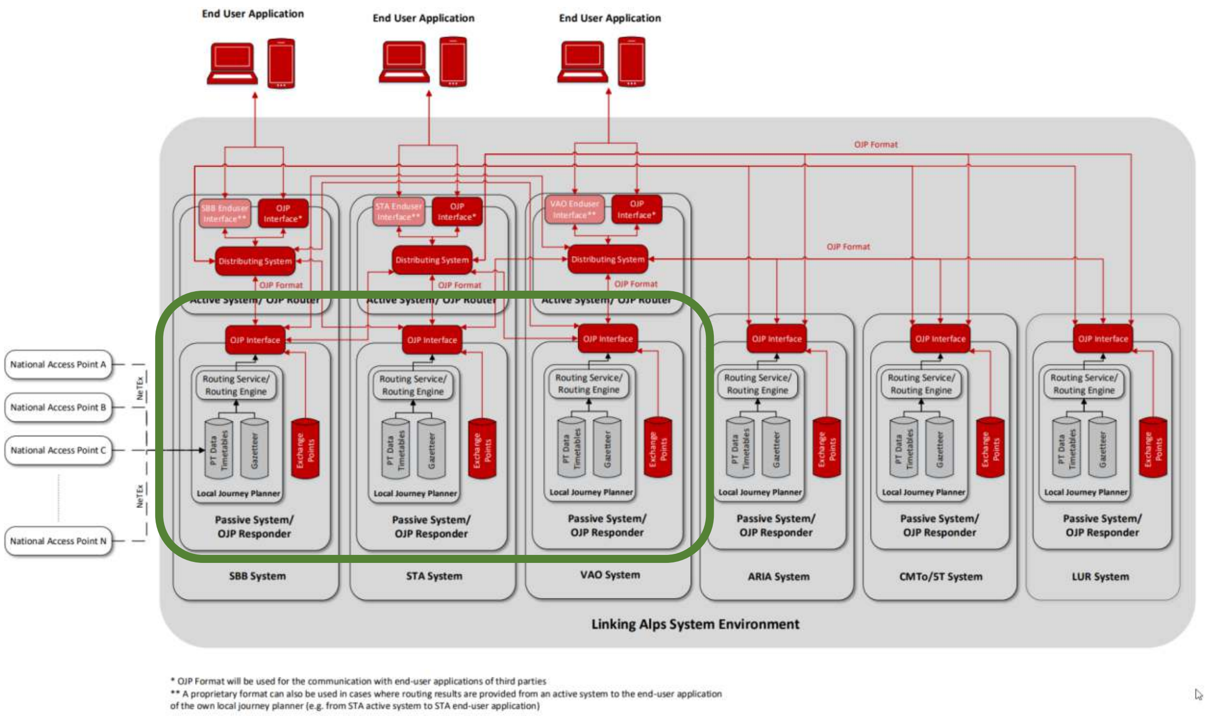


Figure 9: Parts of the LinkingAlps system environment tackled by the interaction between OJP router and OJP responder

4.1.1 Content and physical exchange of the openAPI/OJP interface

A web-based communication network between the participating systems needs to be established as the systems are physically remote. So the the network of journey planners will exchange information via an (web-based) OJP interface fully compliant with the Technical Specification CEN/TC 17118:2018 for "Open API for Distributed Journey Planning". Via this OJP interface requests between the participating services will be exchanged. The OJP interface comprises all requests and responses required to fulfil the tasks of distributed journey planning according to the defined request flows. For the physical exchange the SIRI communication procedures are followed. Detailed exchange communication protocols and subscription processes, compression procedure, security and assurance of data are defined in the specification D.T1.5.1 Specification of the API interface.

The content of the messages is as well defined in the LinkingAlps profile specified in D.T1.5.1 Specification of the API interface. The LinkingAlps OJP Profile aims to define a specific subset of (XML) data elements following a clearly arranged structure in accordance with the OJP standard (CEN/TS 17118:2017) and defined using XML schemas. The schemas include all functionalities required for an OJP interface in order to enable communication with the LinkingAlps distributed journey planning system. In this sense the LinkingAlps OJP Profile defines the content and the structure of the information content as well as the physical exchange format. As the compliance of the developed openAPI services in this project with the Del. Reg. 2017/1926 is a premise, also the compliance with CEN/TS 17118:2017 is a premises. If an extension to the technical Specification CEN/TS 17118:2017 is required, the input to the standardisation group for the revision of the TS is needed. The extension shall be compatible and use extension rules from other familiar standards (like DATEX II). Extensions and rules for extensions need to be included in the LinkingAlps are defined in the LA OJP Profile. Currently there is only one extension: Emissions / CO2. That is handled in the Profile. Extensions need to be handled with care.

4.1.2 Handling of long-distance transport (to calculate the "trunk" leg) & exchange points for trunk leg calculation

If a user requests a trip plan that cannot be responded in a seamless manner through the home system (the local journey planner (LJP), long-distance transport connections are used to connect to neighbouring services (OJP responder). Long-distance transport connection are so called "trunk" legs of the routes that connect at least 2 OJP systems and Exchange points are defined along the "trunk" leg which define which are the neighbouring systems.

The route composition between two local journey planners (home systems) needs a common logic how to link the services by using exchange points. Exchange points are stations that are known by adjacent systems. The definition of exchange points is done in chapter 3.8 Exchange Point Database or Service. That means that the station and the timetable information need to be known by at least by two systems (LJPs).

Trunk leg definition and handling

The so called "trunk" leg are transport connections, either long-distance transport or regional transport connections that interlink journey planners. That means that these trunk legs typically are transnational connections that go beyond the core coverage of the LJPs. Transnational or trans-regional trunk legs are completely provided by one system and not spitted somewhere in between. Consequently, within one trunk leg only one passive system will be tasked with the trip calculation.

All stops along a trunk leg are potential exchange points. All potential exchange points in the LinkingAlps coverage area need to be known by the distributing systems in order to calculate the trunk leg. In detail, all available stops for transnational/trans-regional trunk legs within the system from the system border to the first final stop are potential exchange points and need to be maintained as such in the passive systems. This includes long distance trunks as well as regional trunks. The size of the overall exchange point set has no influence on load or performance behaviour since the set only represents potential possible exchanges. However, the number of considered exchange points for a certain route calculation affects load and performance obviously. The selection of potential relevant exchange points for a certain route is task of the active node, since on this level a kind of global knowledge is available.

Decentral handling of exchange points

That requires that each passive system must be able to mark/flag certain stations as exchange points in their local database. The exchange points must be identified with a system-wide stable and unique

ID. The exchange points are then queried (all at once, or by certain parameters) by other systems on the run. That means that the exchange points are held decentral in the passive systems with the advantage of having the highest actuality.

Handling of exchange point requests

An exchange point service needs to be implemented in the OJP interface. The input data within the exchange point request is the reference to the adjacent system(s) between which the exchange points are to be retrieved (optional). The output data represents a list of all exchange points this responding system has knowledge of (maybe filtered by some adjacent systems specified in the request), their identifiers, names and coordinates and other necessary attributes.

Methods of trunk calculation

In principle, there are different variants how this "trunk" legs are gathered depending on the system architecture settings. Theoretically the trunk legs can be available at the LJPs or they can be available in an Europe-wide trunk timetable information service. In LinkingAlps the trunk legs are held within the LJPs, i.e. held with the passive systems, as there will be no special European-wide transnational router available. Consequently, active systems, in particular the distributing systems, have to be able to either hold the trunk legs in the passive systems or to request the trunk legs from the passive systems before it can start to request for the partial trips. Meaning that there are two different approaches to gather the exchange points needed for the route calculation. Both opportunities shall be enabled through the LinkingAlps system architecture and request flow, leaving the choice of the approach to the active systems. In particular, for the processing of the exchange points within an active system, there is a static and a dynamic approach to get the exchange points from the passive systems:

1) Dynamic approach:

In the dynamic approach, the EPs are requested *for a specific origin-destination-relation (area to stop or stop to area)*. The exchange point request must include and define the “adjacent system” and “destination system”. The passive systems have to provide an estimated travel time to the origin/destination, via the TravelDurationEstimates field.

2) Static approach:

In the static approach, *all* exchange points of a passive system are requested. The exchange point request does not include and define the “adjacent system” and “destination system”. It triggers the passive systems to deliver all exchange points. TravelDurationEstimates are not provided in the exchange point response of passive systems, because the calculation of the travel duration takes place on active system level. Refer to the request flow shown in Figure 11: Sequence diagram, static approach of collecting exchange points knowledge, adapted from a meeting (source VAO).

In case of an *adjacent use case*, there are common exchange points between two passive systems, the route calculation can be done along the trunk leg within one journey planner. Depending on departure or arrival time the overall route is calculated with one multipoint trip request at the origin- or destination system and several 1:1 requests in the correspondent system. Refer to the request flow shown in 3.8 Exchange Point Database or Service.

In the *remote use case,* no common exchange point (or only insufficient ones or from the geography others must be considered) can be identified between two passive systems. This requires the route calculation to hop over one or more passive systems ("hopping"). In this case several multipoint trip requests starting from the origin system respectively the destination system are needed and completed with 1:1 requests.

The precondition to allow that system hopping is the knowledge of all exchange points in the coverage areas within a distributing system, in order to calculate a "meta graph", which is the first step in the distribution algorithm to identify the relevant passive systems and exchange points in order to identify suitable routes. Consequently, the static approach to request exchange points periodically from all system is applied and using the estimated travel duration for the "meta graph", which is a first assumption of the trip duration. This information is utilized because the routing algorithm needs a special logic for filtering and selecting exchange points in order to discard suboptimal routes respectively to reduce the load. This filtering is essential and should be applied for each sub-route (hop). A further objective to reduce performance issues is the minimization of hops since they are quite expensive related to computational power. This first estimate of the meta graph is used to size down the number of potential exchange points to be used for the trip calculation.

4.1.3 Handling of exchange points & relevant requests

As already defined in chapter 3.8 the exchange points (stops or stations) must be known to all participating systems (active and passive). Exchange points can be looked up from a server by using the exchange points service (*ExchangePointsRequest)*.

Figure 10 illustrates details of the two relevant sets of exchange points (VAO-ARIA-CMTo/5T and VAO-SBB-CMTo/5T) for the journey between Vienna and Turin. The exchange points are eventually used for the Trip Request and subsequent MultiPoint Trip Requests.

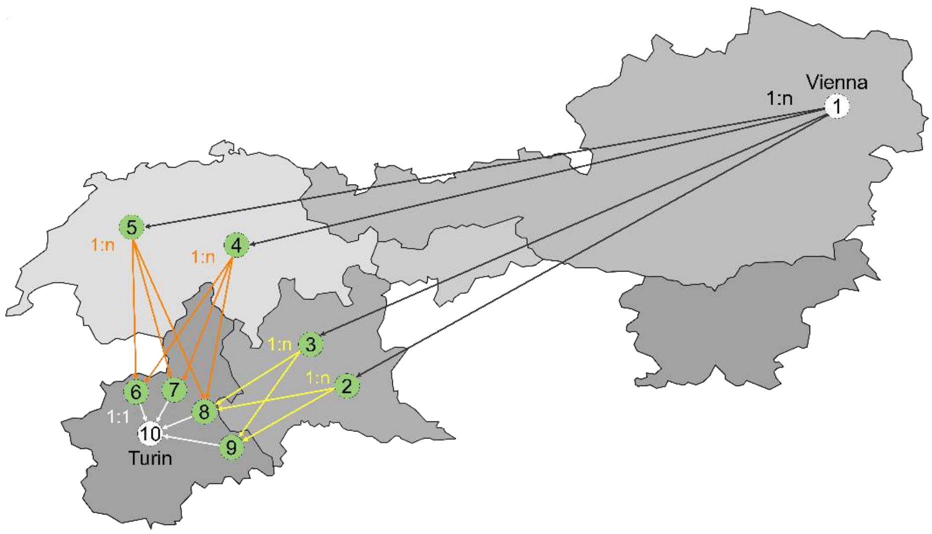


Figure 10: Example: Vienna-Turin, adapted from a meeting (source VAO).

Both the OJP schema and the passive systems shall support a "static" and "dynamic" approach for requesting Exchange Points as described in the previous chapters. A distributing system can obtain exchange points for distributed route calculation with a static or dynamic approach. A "static approach" implies that all information on exchange points is gathered and stored periodically and a dynamic approach implies that exchange points are collected just before trip request is made. Figure 11 shows the static approach on exchange point knowledge collection.

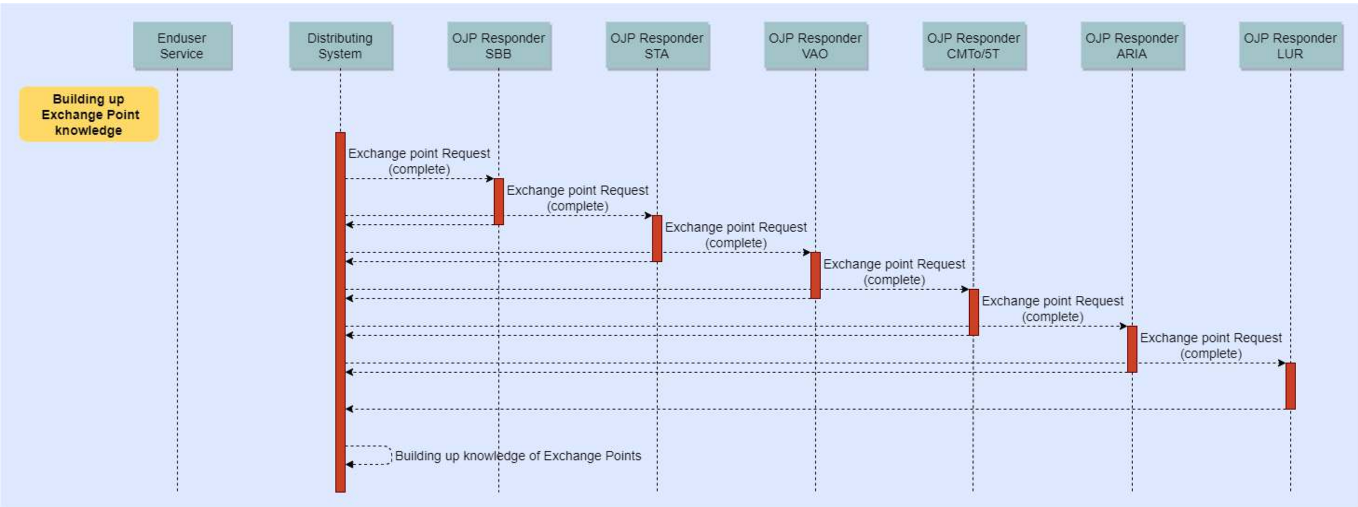


Figure 11: Sequence diagram, static approach of collecting exchange points knowledge, adapted from a meeting (source VAO).

With the knowledge of all exchange points a Multipoint Trip can be planned using a distributed route calculation (Example: journey Vienna-Turin), as can be seen in the sequence diagram on the Figure 12.

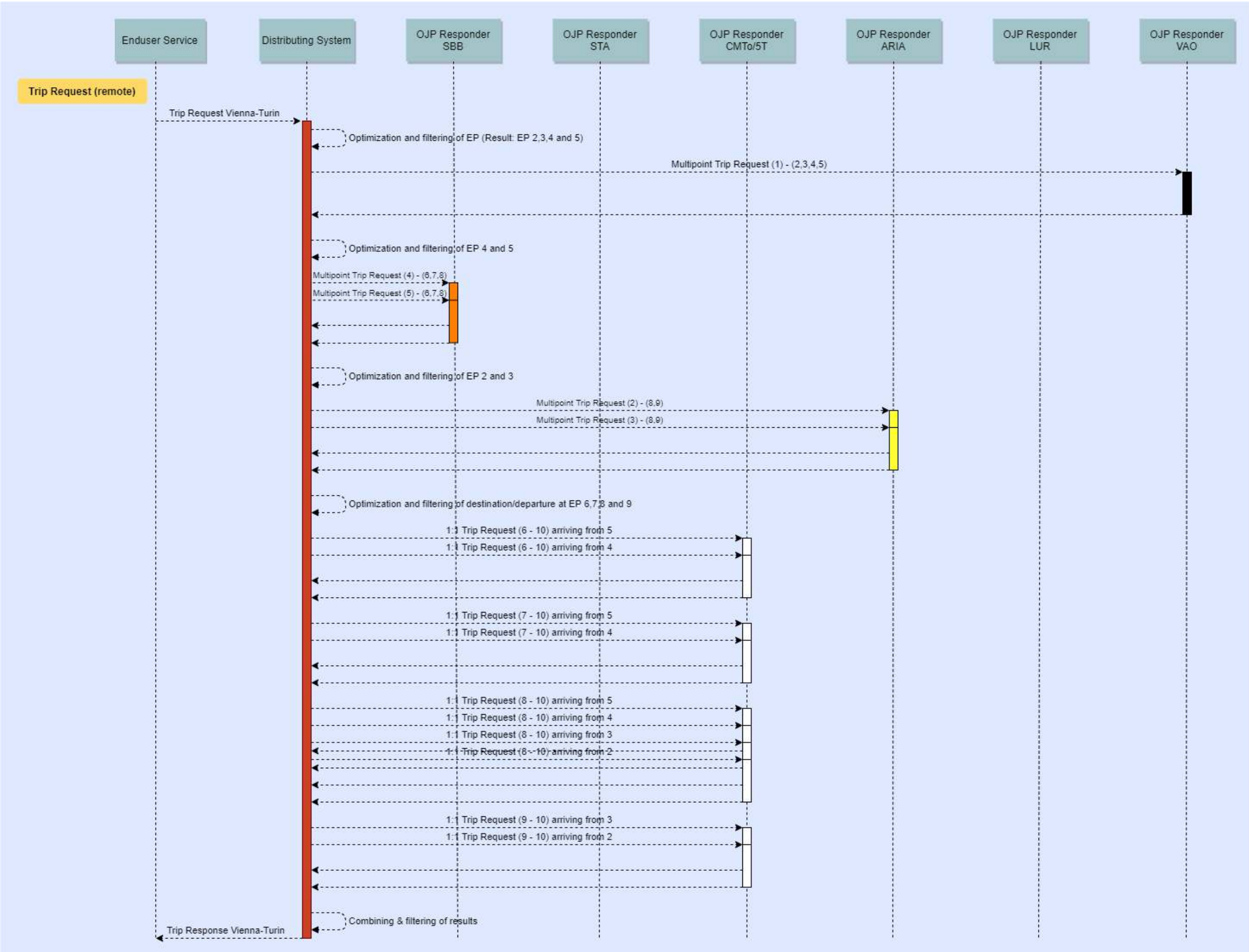


Figure 12: Sequence diagram, Vienna-Turin, fully distributed route calculation, adapted from a meeting (source VAO).

The following tables (Table 13- Table 16) were extracted from the CEN/TS 17118:2017 Public transport — Open API for distributed journey planning and they describe request and response structures for exchange points.

Table 13: Description of *ExchangePointsRequestStructure, adapted from CEN/TS 17118:2017 Public transport — Open API for distributed journey planning*.

ExchangePointsRequestStructure			*+Structure*	Container for exchange points request details.
ExchangePointsRequest	*PlaceRef*	0:1	*+PlaceRef*	Location for which exchange points to other "neighbour" systems are to be searched. This location is usually the origin/destination of a passenger trip. May be omitted if all exchange points shall be returned.
	Params	0:1	*+ExchangePointsParam*	More parameters for restricting the request (Table "Description of *ExchangePointsParamStructure*").
	Extension	0:1	*xs:anyType*	Extensions.

Table 14: Description of *ExchangePointsParamStructure, adapted from CEN/TS 17118:2017 Public transport — Open API for distributed journey planning*.

ExchangePointsParamStructure			*+Structure*	Contains the parameters controlling the search for exchange points.
ExchangePointsDataFilter	*Type*	0:*	*stop* | *address* | *poi* | *coord* | *TopographicPlace*	Allowed location object types. If none is given, all types are allowed.
	Usage	0:1	origin | destination | via	Defines, whether location objects for origin, via, or destination are searched.
	PtModes	0:1	*+PtModeFilter*	Allowed public transport modes. Defines, which public transport modes have to be available at the returned location objects. Applies only to stops.
	OperatorFilter	0:1	*+OperatorFilter*	Filter for locations that are opened by certain organisations.
	TopographicPlaceRef	0:*	*→TopographicPlaceCode*	If at least one is set, only location objects within the given localities are allowed.
	DestinationSystem	0:1	*siri:ParticipantRef*	Reference to system in which the destination (or origin) of the passenger is located.
	AdjacentSystem	0:*	*siri:ParticipantRef*	One or more adjacent systems for which the exchange points should be retrieved.
ExchangePointsPolicy	*Language*	0:1	*xs:language*	Preferred language in which to return text values.
	NumberOfResults	0:1	*xs:positiveInteger*	Maximum number of results to be returned. The service is allowed to return fewer objects if reasonable or otherwise appropriate. If the number of matching objects is expected to be large (eg: in the case that all objects should be delivered) this parameter can be used to partition the response delivery into smaller chunks. The location information service is expected to support a response volume of at least 500 location objects within one single response.
	ContinueAt	0:1	*xs:nonNegativeInteger*	Tells the server to skip the mentioned number of results in its response. Can be used in a follow-up request to get further results. The value is usually taken from the previous response (element *ContinueAt, see Table "Description of ExchangePointsResponseStructure"*) .To request the remaining objects the request is repeated with *ContinueAt* set to the value given by the last response.

An element *ExchangePointsResponse* of the type *ExchangePointsResponseStructure* is used to respond to an exchange points request.

Table 15: Description of *ExchangePointsResponseStructure, adapted from CEN/TS 17118:2017 Public transport — Open API for distributed journey planning*.

ExchangePointsResponseStructure			*+Structure*	Container for the results of an exchange points query.
	ErrorMessage	0:*	*+ErrorMessage*	Error messages related to the processing of the entire request.
ExchangePointsResponse	*ContinueAt*	0:1	*xs:nonNegativeInteger*	If this value is set the service indicates that there are more location objects to be returned than one response can transmit (due to size limits). The value of *ContinueAt* can be used in a follow-up request to get further results. It tells the server to skip the given number of results in its response.
	Place	0:*	*+ExchangePointsResult*	The Exchange points found by the service (see Table "Description of *ExchangePointsResultStructure*").

Table 16: Description of *ExchangePointsResultStructure*, adapted *from CEN/TS 17118:2017 Public transport — Open API for distributed journey planning*.

ExchangePointsResultStructure			*+Structure*	Result structure for an exchange point object.
	Place	1:1	*+Place*	Place object that describes this exchange point.
	TravelDurationEstimate	0:1	*xs:duration*	Rough estimate of the travel duration from the specified place to this exchange point.
	BorderPoint	0:1	*xs:Boolean*	Flag if this exchange point is an administrative border point where timetables are cut off while services still may run through and connect the regions. At this kind of point passengers may continue their trip on the same service. Default is FALSE.
	Mode	0:*	*+Mode*	List of transport modes that call at this location object. This list should only be created in case of stop points or stop places – and only when explicitly requested.

4.1.4 Supported requests, flow and sequence – demanded by the OJP routing and orchestration of messages

The OJP router is a home system (local journey planning system) with the capability of distributed journey planning, meaning the route composition algorithm exists that is able to link together trip legs coming from the home system and other journey planning systems. In the following, it will be referred to as OJP routing algorithm. The active systems will provide the OJP routing capabilities and the distributing system that is facilitating the trip leg enquiries through the OJP interface. Therefore, the active systems are referred to as OJP router, meaning that they include an OJP routing and distributing system.

Each OJP router (active system) can supply its own individual OJP routing algorithm based on its particular demands (e.g. performance). This means that the routing results calculated by different OJP routers is not equal for the same trip request.

In order to facilitate the distributed route composition with the OJP routing algorithm the required trip legs and supporting data need to be requested over the openAPI for distributed journey planning (openAPI). The distribution of the requests is facilitated by a distributing system. The orchestration of the messages is also part of the tasks of the distributing system and depend on the used algorithm.

The interoperability aspect within the used OJP routing lies within the needed requests and the sequence of the request that need to be distributed by the distributing system. That means the orchestration of the messages determines the sequence of the messages and probably also the content of the messages (e.g. to need to deliver all stops and all exchange points in one run).

Therefore, the network of journey planners, OJP router and OJP responder, need to support the request and the sequence. Considering that different OJP routing might have different trip composition mechanism the complexity will raise with the number of different OJP routings. **A harmonisation of the OJP routing (orchestration algorithm) is therefore recommended. If this is not possible, due to diverging approaches, the details of the request flows need to be published and aligned among all system parts.**

Whatever sequence is chosen for the OJP routing both forward and backward planning will be necessary in order to allow trip optimisation.

4.2 Interaction of OJP router and end user application

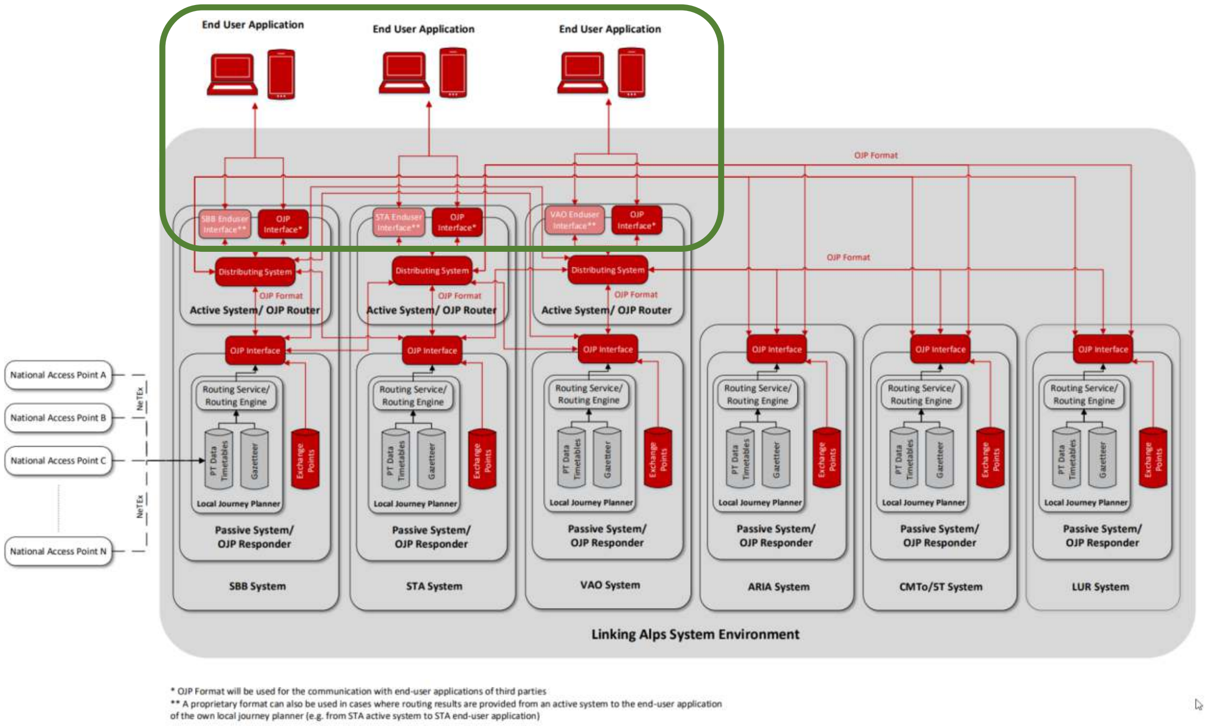


Figure 13: Parts of the LinkingAlps system environment tackled by the interaction between OJP router and end user application

This chapter describes the modalities with which the OJP router and an end user application shall exchange distributed journey planning information. Through the end-user application an end user is able to make trip requests (e.g. by providing the start and end point, as well as other trip options) and receive and visualize the results in a properly formatted manner, e.g. on a map. In general, an end user application can be implemented by a third party organization, which is different from the one that makes at disposal the DRJP service.

Workflow: the reference workflow for the interaction between OJP router and end-user application is the following:

- through a certain GUI (typically over a map and/or a parameter input window) the end user is able to set up all available parameters for the trip planning
- The geolocations are selected (that can be done by auto-complete and selection lists. The geolocations (mostly stops) are selected by LocationInformationRequest/LocationInformationResponse
- When with this selection the information on the trip is completed and submitted by the end user, the end-user application sends the request to the OJP router, preferably over the OJP interface (TripRequest)

- once computed, the end-user application receives back the trip information from the OJP router, preferably over OJP interface (TripResponse)
- the end-user application presents the trip calculation results in the GUI, e.g. as trip detail information and route on the map
- for later requests during the trip, the app can use the selected trip and inqury updates with (TripInformationRequest/TriInformationResponse)

Precondition: the end-user application is entitled to access the OJP router according to the terms and conditions defined by the owner of the DRJP service, which are defined according to the organisational requirements set below.

4.3 Authorisation, authentication and compression of the network

Token-based authentication is used to grant access to the API when a client is making API calls on the OJP interface. An API key is a token that is provided in the request header and or the query string or in the body. In LinkingAlps it is used in the header, and can be used in the body of the request. The API keys are used between

- end-user-application (client) and active system (server)
- the active system (client) and the passive system (server)

The API key is considered to be used by functional user (end user applications) and not persons. A authentication of users is not defined.

As the key are supposed to be secret and need to be protected from third-party access in transmission from client to server. Therefore, security mechanism HTTPS + TLS1.3 is applied for encryption and identification of client and server (handshake mechanisms).

4.4 Service and data quality

The quality of the final service and data is relying on the quality of the participating service. It needs basic quality criteria (minimum criteria) for the participation of systems. A guideline could be the EU-EIP Multimodal Travel Information Services (MMTIS): Update of the Quality Framework.

Service-Level-Agreements (SLAs) and Operational Level Agreements (OLA) might be used to formulate the commitment between a service provider and a client. In general, in a SLA particular aspects of the service – quality, availability, responsibilities – are agreed between the parties.

5 Organisational Requirements (part of WPT3)

In addition to the technical requirements a range of essential organisational requirements were identified, which complement the technical architecture and are crucial for an efficient and smooth operation of the LinkingAlps distributed journey planning system.

Organisational requirements comprise one side organisational processes which provide an organisational scaffolding for technical processes. These organisational processes are necessary for a flawless integration of amended and new technical requirements into existing systems as well as to ensure an effective operation of the LinkingAlps service. Organisational processes are necessary on different systemic levels, and cover processes on system(s) level, on LinkingAlps network-wide (global) level as well as are necessary for an innovation of the LinkingAlps network.

While processes on system(s) level cover exchange point matching and update processes, API-key handling processes, support and incident handling processes as well as end-user app integration processes, network-wide (global) organisational processes are mainly developed to cover administrative and coordinative activities as well as network-wide support activities which include issues such as cyber security and attack mitigation on a global level. Furthermore, the need to elaborate organisational processes for the innovation of the LinkingAlps network became also apparent. Thus, organisational processes for the admission of new systems or new end-user applications into the LinkingAlps network were elaborated, processes for the upgrade of passive systems to actives system are covered as well as overall innovation coordination process established.

On the other side, the organisational architecture also discusses necessary regulative structures, which provide the foundation for licencing and agreement matters. Overall, key aspects which are essential to be stipulated and legally agreed upon are identified and summarised as regulative requirements. Although, the final agreements need to be bilaterally negotiated, between the relevant systems, the regulative requirements provide an overview of the LinkingAlps regulative structure.

The organisational architecture, which includes the above mentioned organisational processes as well as the regulative structure, is part of WP T3 and was developed in Activity 3.2 and thus, a detailed elaboration can be found in Deliverable 3.2.1.