



BSI Standards Publication

Intelligent transport systems — Public transport — Open API for distributed journey planning

National foreword

This Published Document is the UK implementation of CEN/TS 17118:2024. It supersedes PD CEN/TS 17118:2017, which is withdrawn.

The UK participation in its preparation was entrusted to Technical Committee EPL/278, Intelligent transport systems.

A list of organizations represented on this committee can be obtained on request to its committee manager.

Contractual and legal considerations

This publication has been prepared in good faith, however no representation, warranty, assurance or undertaking (express or implied) is or will be made, and no responsibility or liability is or will be accepted by BSI in relation to the adequacy, accuracy, completeness or reasonableness of this publication. All and any such responsibility and liability is expressly disclaimed to the full extent permitted by the law.

This publication is provided as is, and is to be used at the recipient's own risk.

The recipient is advised to consider seeking professional guidance with respect to its use of this publication.

This publication is not intended to constitute a contract. Users are responsible for its correct application.

This publication is not to be regarded as a British Standard.

© The British Standards Institution 2024
Published by BSI Standards Limited 2024

ISBN 978 0 539 30043 7

ICS 35.240.60

Compliance with a Published Document cannot confer immunity from legal obligations.

This Published Document was published under the authority of the Standards Policy and Strategy Committee on 31 December 2024.

Amendments/corrigenda issued since publication

Date	Text affected

TECHNICAL SPECIFICATION
SPÉCIFICATION TECHNIQUE
TECHNISCHE SPEZIFIKATION

CEN/TS 17118

December 2024

ICS 35.240.60

Supersedes CEN/TS 17118:2017

English Version

**Intelligent transport systems - Public transport - Open API
for distributed journey planning**

Intelligente Verkehrssysteme - Öffentlicher Verkehr -
Offene API für verteilte Reiseplanung

This Technical Specification (CEN/TS) was approved by CEN on 26 May 2024 for provisional application.

The period of validity of this CEN/TS is limited initially to three years. After two years the members of CEN will be requested to submit their comments, particularly on the question whether the CEN/TS can be converted into a European Standard.

CEN members are required to announce the existence of this CEN/TS in the same way as for an EN and to make the CEN/TS available promptly at national level in an appropriate form. It is permissible to keep conflicting national standards in force (in parallel to the CEN/TS) until the final decision about the possible conversion of the CEN/TS into an EN is reached.

CEN members are the national standards bodies of Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Republic of North Macedonia, Romania, Serbia, Slovakia, Slovenia, Spain, Sweden, Switzerland, Türkiye and United Kingdom.



EUROPEAN COMMITTEE FOR STANDARDIZATION
COMITÉ EUROPÉEN DE NORMALISATION
EUROPÄISCHES KOMITEE FÜR NORMUNG

CEN-CENELEC Management Centre: Rue de la Science 23, B-1040 Brussels

© 2024 CEN All rights of exploitation in any form and by any means reserved
worldwide for CEN national Members.

Ref. No. CEN/TS 17118:2024 E

Contents

	Page
European foreword.....	12
0 Introduction	13
0.1 General.....	13
0.2 An Open API for distributed journey planning (OJP).....	13
0.3 The public transport information tensions	13
0.4 Distributed journey planning architecture beyond scope.....	14
0.4.1 General.....	14
0.4.2 The distributed journey planning approach.....	14
0.4.3 Distributed or centralised approaches	15
0.4.4 The basis for the Open API.....	15
0.4.5 Other possible uses for the Open API.....	16
0.5 The European ITS Directive	16
1 Scope	17
2 Normative references	17
3 Terms and definitions	17
3.1 Terms used in OJP schema.....	17
3.2 Terms in OJP Schema	33
4 Symbols and abbreviations	34
5 Use cases.....	35
5.1 General.....	35
5.2 Key tasks for Distributed Journey Planning.....	37
5.2.1 Planning of a trip	37
5.2.2 Discovering relevant stops	38
5.2.3 Obtaining information about accessibility and services for those with special needs.....	38
5.2.4 Seeking route information that can be displayed on maps	38
5.2.5 Refine a trip.....	38
5.2.6 Changing a Trip.....	39
5.3 Additional tasks for a Distributed Journey Planning system.....	39
5.3.1 Requesting a stop timetable.....	39
5.3.2 Requesting times and other information for all intermediate stops in a trip.....	40
5.3.3 Requesting expected events at a particular stop.....	40
5.3.4 Requesting information about a given journey or vehicle.....	40
5.3.5 Requesting information about the fares and ticket options for a particular trip.....	40
5.3.6 Requesting information about lines.....	40
5.3.7 Requesting information about availability of journeys/vehicles.....	40
5.3.8 Other possible questions.....	41
5.4 Advanced concepts/considerations	41
5.4.1 Routing for passengers with special needs.....	41
5.4.2 Via, fixed legs	42
5.4.3 User preferences	43
5.4.3.1 Definition	43
5.4.3.2 Usage.....	43
5.4.3.3 Good user preferences	43
5.4.3.4 Mechanism.....	44
5.4.3.5 Suggested user preferences.....	44

5.4.4	IncludeAlternativeOptions in trips	47
5.4.5	Handling of slow traffic	47
5.4.6	Demand responsive transport.....	48
5.4.6.1	Relevant mode	49
5.4.6.2	Handling demand responsive buses in the stop event service	49
5.4.7	Formations, Occupancy and Capacity.....	50
5.4.8	International ServiceJourney (mainly trains).....	51
5.4.9	Interaction OJP with fare calculating systems and reservation systems	52
6	System architectures, metadata and data.....	52
6.1	General	52
6.2	General considerations – distributed planning.....	52
6.3	Modular interface construction.....	54
6.4	General considerations – pipelines.....	55
6.5	Responsibility of a Distributing System to find gaps/overlaps in the results and to correct them	55
6.6	Metadata requirements.....	55
6.7	Core data requirements	56
6.7.1	Locations.....	56
6.7.2	Topographic map.....	57
6.7.3	Timetables.....	57
6.7.4	Fares and booking information	57
6.7.5	Other modes	57
7	Open API for distributed journey planning – OJP services.....	58
7.1	Identification of objects beyond system borders.....	58
7.1.1	General	58
7.1.2	Stops and Stopping Points	58
7.1.3	Localities and Districts	59
7.1.4	Addresses and POIs.....	59
7.1.5	Organisations: transport companies and transport authorities.....	59
7.1.6	Lines and line directions.....	60
7.1.7	Journeys	61
7.1.8	Vehicles	61
7.1.9	Operating days.....	61
7.1.10	Owners.....	61
7.1.11	Stop and vehicle equipment.....	62
7.1.12	Participating systems or IT systems.....	62
7.1.13	Incident messages	62
7.1.14	Fare authority	62
7.1.15	Tariff zones	62
7.1.16	Tickets and traveller cards	62
7.2	Trip service	63
7.2.1	Purpose	63
7.2.2	Interactions.....	64
7.2.3	Concerned components.....	65
7.2.4	Function 1: trip planning	65
7.2.5	Function 2: multipoint trip planning.....	65
7.2.6	Function 3: distributed trip planning.....	67
7.2.7	Function 4: find earlier or later trips	67
7.3	Departure/arrival monitor	67
7.3.1	Purpose	67
7.3.2	Interactions.....	67
7.3.3	Concerned components.....	68

7.3.4 Function 5: departure/arrival monitor 68

7.4 Fare information 68

7.4.1 Purpose..... 68

7.4.2 Interactions..... 68

7.4.3 Concerned Components..... 69

7.4.4 Function 6: tariff zones for stop or station 69

7.4.5 Function 7: static fare information..... 69

7.4.6 Function 8: trip-related fare information 70

7.5 Location text matching..... 70

7.5.1 Purpose..... 70

7.5.2 Interactions..... 70

7.5.3 Concerned Components..... 70

7.5.4 Function 9: Location text matching 70

7.6 Object information service 71

7.6.1 Purpose..... 71

7.6.2 Interactions..... 71

7.6.3 Concerned components 71

7.6.4 Function 10: Object Information 71

7.6.5 Function 11: finding relevant exchange points..... 71

7.7 Trip Information Service..... 72

7.7.1 Purpose..... 72

7.7.2 Interactions..... 72

7.7.3 Concerned components 72

7.7.4 Function 12: trip information service 72

7.8 Availability service 73

7.8.1 Purpose..... 73

7.8.2 Interactions..... 73

7.8.3 Concerned Components..... 73

7.8.4 Function 13: availability service..... 74

7.9 Refinement service..... 74

7.9.1 Purpose..... 74

7.9.2 Interactions..... 75

7.9.3 Concerned Components..... 75

7.9.4 Function 14: trip refinement service 75

7.10 Trip changing service 76

7.10.1 Purpose..... 76

7.10.2 Interactions..... 76

7.10.3 Concerned Components..... 76

7.10.4 Function 15: trip changing service 76

7.11 Line information service 77

7.11.1 Purpose..... 77

7.11.2 Interactions..... 77

7.11.3 Concerned components 77

7.11.4 Function 16: line information service 77

7.12 Status service..... 78

7.12.1 Purpose..... 78

7.12.2 Interactions..... 78

7.12.3 Concerned components 78

7.12.4 Function 17: status service 78

7.13 When to use which service..... 79

8 Open API for distributed journey planning – interface description..... 80

8.1 Notation of XML elements and XML structures 80

8.1.1	General	80
8.1.2	Display of XML elements in the text.....	81
8.1.3	Display of Relationships.....	81
8.1.4	Table notation of XML structures	81
8.1.4.1	General	81
8.1.4.2	Grouping	85
8.1.4.3	Element name	85
8.1.4.4	Multiplicity & choice (min:max).....	85
8.1.4.5	Data type	85
8.1.4.6	Explanation	86
8.1.5	Message exchange	86
8.1.6	Use of SIRI procedure.....	86
8.1.7	HTTP and REST.....	87
8.1.8	Roles of server and client	88
8.2	Services and XML schemas.....	88
8.2.1	General	88
8.2.2	Services provided	88
8.2.3	Service sequences.....	89
8.2.4	Imported schemas.....	90
8.2.5	Problems and error states when operating OJP services.....	90
8.2.6	Error codes from SIRI.....	91
8.2.7	OJP <i>ErrorCondition</i>	92
8.2.8	General OJP problems.....	92
8.2.9	Time zones	92
8.3	Common XML structures.....	93
8.3.1	General	93
8.3.2	Root element OJP	93
8.3.2.1	General	93
8.3.2.2	OJPRequestStructure.....	94
8.3.2.3	OJPResponseStructure.....	95
8.3.2.4	ServiceRequestStructure and ServiceRequestContext (from SIRI).....	96
8.3.2.5	ServiceDeliveryStructure (from SIRI).....	100
8.3.3	OJP_Utility.....	101
8.3.3.1	General	101
8.3.3.2	Simple types.....	101
8.3.3.3	InternationalTextStructure	101
8.3.3.4	WebLinkStructure	102
8.3.4	OJP_ModesSupport.....	102
8.3.4.1	General	102
8.3.4.2	Simple types.....	102
8.3.4.3	IndividualTransportOptionStructure	107
8.3.4.4	ItModesStructure	108
8.3.4.5	ModeStructure	108
8.3.4.6	ModeAndModeOfOperationFilterStructure	109
8.3.4.7	ModeFilterStructure	110
8.3.5	OJP_Common	111
8.3.5.1	Simple types.....	111
8.3.5.2	OJPError.....	111
8.3.5.3	OJPErrorStructure.....	112
8.3.5.4	ErrorType	112
8.3.5.5	PrivateCodeStructure.....	112
8.3.5.6	LinearShapeStructure.....	112
8.3.5.7	AreaStructure.....	112

8.3.5.8 OperatorRef.....	112
8.3.5.9 OperatorRefs_RelStructure	112
8.3.5.10 OperatorFilterStructure.....	113
8.3.5.11 ProductCategoryRef.....	113
8.3.5.12 siri:LineDirectionStructure.....	113
8.3.5.13 LineDirectionFilterStructure.....	113
8.3.5.14 JourneyRefStructure.....	113
8.3.5.15 JourneyRef.....	113
8.3.5.16 VehicleFilterStructure	114
8.3.5.17 AlternativeServiceStructure.....	114
8.3.5.18 OwnerRefStructure	114
8.3.5.19 OwnerRef.....	114
8.3.5.20 OperatingDayRefStructure.....	114
8.3.5.21 OperatingDayRef.....	114
8.3.5.22 OperatingDaysStructure	115
8.3.5.23 WeekdayTimePeriodStructure.....	115
8.3.5.24 GeneralAttributeStructure.....	115
8.3.5.25 EmissionCO2Structure.....	115
8.3.6 OJP_PlaceSupport.....	116
8.3.6.1 General.....	116
8.3.6.2 Simple types	116
8.3.6.3 StopPointStructure.....	117
8.3.6.4 StopPlaceRefStructure	117
8.3.6.5 StopPlaceRef.....	117
8.3.6.6 StopPlaceStructure.....	117
8.3.6.7 TopographicPlaceRefStructure.....	118
8.3.6.8 TopographicPlaceRef.....	118
8.3.6.9 TopographicPlaceStructure	118
8.3.6.10 PointOfInterestRefStructure	118
8.3.6.11 PointOfInterestRef	118
8.3.6.12 PointOfInterestStructure	119
8.3.6.13 PointOfInterestCategoryStructure	119
8.3.6.14 PointOfInterestAdditionalInformationStructure	119
8.3.6.15 CategoryKeyValue Type	119
8.3.6.16 OsmTagStructure.....	120
8.3.6.17 PointOfInterestFilterStructure.....	120
8.3.6.18 AccessModesListOfEnumerations.....	120
8.3.6.19 AddressRefStructure	120
8.3.6.20 AddressRef.....	120
8.3.6.21 AddressStructure.....	120
8.3.6.22 PlaceStructure.....	121
8.3.6.23 PlaceRefStructure.....	121
8.3.6.24 LocationProblemType.....	122
8.3.6.25 ExchangePointsProblemType	122
8.3.7 OJP_JourneySupport.....	122
8.3.7.1 General.....	122
8.3.7.2 Simple types	122
8.3.7.3 ServiceViaPointStructure.....	123
8.3.7.4 ProductCategoryStructure.....	123
8.3.7.5 TripViaStructure	123
8.3.7.6 ParallelServiceStructure	124
8.3.7.7 DatedJourneyStructure.....	126
8.3.7.8 TripLocationStructure	129

8.3.7.9 ServiceArrivalStructure	130
8.3.7.10 ServiceDepartureStructure	130
8.3.7.11 CallAtStopStructure	131
8.3.7.12 ContinuousServiceStructure	131
8.3.7.13 VehiclePositionStructure	135
8.3.7.14 ProgressBetweenStopsStructure	135
8.3.7.15 PlaceContextStructure	135
8.3.7.16 LegAttributeStructure	136
8.3.7.17 LegTrackStructure	136
8.3.7.18 TrackSectionStructure	136
8.3.8 OJP_FacilitySupport	136
8.3.8.1 General	136
8.3.8.2 siri:CommonFacilityGroup	137
8.3.8.3 siri:StopFacilityGroup	138
8.3.8.4 siri:ServiceFacilityGroup	138
8.3.8.5 siri:AllFacilitiesGroup	138
8.3.9 OJP_SituationSupport	139
8.3.9.1 General	139
8.3.9.2 SituationsStructure	139
8.3.9.3 SituationFullRef	139
8.3.9.4 SituationFullRefStructure	139
8.3.9.5 SituationRefList	140
8.3.10 OJP_RequestSupport	140
8.3.10.1 General	140
8.3.10.2 Simple types	140
8.3.10.3 AbstractOJPServiceRequestStructure	141
8.3.10.4 OJP delivery structures – PlacesStructure	141
8.3.10.5 OJP delivery structures – Operators_RelStructure	141
8.3.10.6 OJP delivery structures – ResponseContextStructure	141
8.3.10.7 OJP delivery structures – OJPGenericProblemType	141
8.3.10.8 OJP DeliveryStructure template & AbstractServiceDeliveryStructure (from SIRI) ..	142
8.3.11 OJP_FareSupport	144
8.3.11.1 General	144
8.3.11.2 Simple types	144
8.3.11.3 BookingNotesStructure	147
8.3.11.4 FareAuthorityRefStructure	147
8.3.11.5 FareAuthorityRef	147
8.3.11.6 TariffZoneRefStructure	147
8.3.11.7 TariffZoneRef	147
8.3.11.8 TariffZoneStructure	147
8.3.11.9 TariffZoneListInAreaStructure	147
8.3.11.10 TariffZoneRefListStructure	148
8.3.11.11 ContactDetailsStructure	148
8.3.11.12 BookingMethodListOfEnumerations	148
8.3.11.13 PurchaseMomentListOfEnumerations	148
8.3.11.14 BookingProcessListOfEnumerations	148
8.3.11.15 GroupBookingListOfEnumerations	148
8.3.11.16 BookingArrangementsStructure	148
8.3.11.17 BookingArrangementsContainerStructure	149
8.3.11.18 FareProductRefStructure	149
8.3.11.19 FareProductRef	149
8.3.11.20 EntitlementProductListStructure	149
8.3.11.21 EntitlementProductStructure	150

8.3.11.22	FareProductStructure	150
8.3.11.23	TripFareResultStructure	151
8.3.11.24	FarePassengerStructure	151
8.3.11.25	FareParamStructure	152
8.3.11.26	FareProblemType	152
8.4	Service location information	152
8.4.1	Description	152
8.4.2	Simple types	153
8.4.3	Request structures	153
8.4.3.1	General	153
8.4.3.2	LocationInformationRequestStructure	153
8.4.3.3	InitialLocationInputStructure	154
8.4.3.4	GeoRestrictionsStructure	154
8.4.3.5	GeoCircleStructure	154
8.4.3.6	GeoRectangleStructure	154
8.4.3.7	GeoAreaStructure	155
8.4.3.8	PlaceParamStructure	155
8.4.4	Delivery structures	156
8.4.4.1	General	156
8.4.4.2	LocationInformationDeliveryStructure	156
8.4.4.3	PlaceResultStructure	157
8.5	Service Exchange Points	157
8.5.1	Description	157
8.5.2	Request Structures	158
8.5.2.1	General	158
8.5.2.2	OJPEXchangePointsRequestStructure	158
8.5.2.3	ExchangePointsParamStructure	158
8.5.3	Delivery Structures	159
8.5.3.1	General	159
8.5.3.2	OJPEXchangePointsDeliveryStructure	159
8.5.3.3	ExchangePointsResultStructure	159
8.6	Service intermodal trip planning	160
8.6.1	Description	160
8.6.2	Distributed planning of intermodal trips	160
8.6.3	Simple types	162
8.6.4	Request structures	168
8.6.4.1	General	168
8.6.4.2	OJPTripRequestStructure	168
8.6.4.3	OJPMultiPointTripRequestStructure	169
8.6.4.4	TripParamStructure	169
8.6.4.5	MultiPointTripParamStructure	172
8.6.4.6	NotViaStructure	174
8.6.4.7	NoChangeAtStructure	175
8.6.4.8	TariffZoneFilterStructure	175
8.6.5	Delivery structures	175
8.6.5.1	General	175
8.6.5.2	OJPTripDeliveryStructure	175
8.6.5.3	OJPMultiPointTripDeliveryStructure	176
8.6.5.4	TripResultStructure	176
8.6.5.5	TripProblemType	176
8.6.5.6	MultiPointTripResultStructure	178
8.6.5.7	TripStructure	178
8.6.5.8	TripSummaryStructure	179

8.6.5.9	LegStructure	180
8.6.5.10	TimedLegStructure	180
8.6.5.11	TransferLegStructure	181
8.6.5.12	ContinuousLegStructure	182
8.6.5.13	LegBoardStructure	182
8.6.5.14	LegAlightStructure	183
8.6.5.15	LegIntermediateStructure	185
8.6.5.16	PathGuidanceStructure	186
8.6.5.17	PathGuidanceSectionStructure	186
8.6.5.18	FollowStructure	187
8.6.5.19	PathLinkStructure	187
8.6.5.20	PathLinkEndStructure	187
8.7	Service stop events (departure/arrival monitor)	187
8.7.1	Description	187
8.7.2	Simple types	188
8.7.3	Request structures	188
8.7.3.1	General	188
8.7.3.2	StopEventRequestStructure	189
8.7.3.3	StopEventParamStructure	189
8.7.4	Response Structures	190
8.7.4.1	General	190
8.7.4.2	OJPStopEventDeliveryStructure	191
8.7.4.3	StopEventResultStructure	191
8.7.4.4	StopEventProblemType	191
8.7.4.5	StopEventStructure	192
8.7.4.6	CallAtNearStopStructure	192
8.8	Service trip information	192
8.8.1	Description	192
8.8.2	Simple types	193
8.8.3	Request structures	193
8.8.3.1	General	193
8.8.3.2	OJPTripInfoRequestStructure	194
8.8.3.3	TripInfoParamStructure	194
8.8.4	Response Structures	195
8.8.4.1	General	195
8.8.4.2	TripInfoDeliveryStructure	195
8.8.4.3	TripInfoResultStructure	195
8.8.4.4	TripInfoProblemType	196
8.9	Service tickets and fare calculation	196
8.9.1	Description	196
8.9.2	Simple types	196
8.9.3	Request structures	196
8.9.3.1	General	196
8.9.3.2	OJPFareRequestStructure	196
8.9.3.3	StopFareRequestStructure	197
8.9.3.4	StaticFareRequestStructure	197
8.9.3.5	TripFareRequestStructure	197
8.9.3.6	MultiTripFareRequestStructure	197
8.9.3.7	PlaceFareRequestStructure	198
8.9.4	Delivery structures	198
8.9.4.1	General	198
8.9.4.2	OJPFareDeliveryStructure	198
8.9.4.3	FareResultStructure	198

8.9.4.4 StopFareResultStructure	199
8.9.4.5 StaticFareResultStructure	199
8.9.4.6 TripFareProductReferenceStructure	200
8.9.4.7 MultiTripFareResultStructure	200
8.9.4.8 PlaceFareResultStructure	200
8.10 Service for refining a trip	200
8.10.1 Description	200
8.10.2 Request structures	200
8.10.2.1 General	200
8.10.2.2 OJPTripRefineRequestStructure	201
8.10.2.3 TripRefineParamStructure	201
8.10.3 Delivery structures	204
8.10.3.1 General	204
8.10.3.2 OJPTripRefineDelivery	204
8.11 Service for changing a trip	205
8.11.1 Request structures	205
8.11.1.1 General	205
8.11.1.2 OJPTripChangeRequestStructure	205
8.11.1.3 TripChangeParamStructure	205
8.11.2 Delivery structures	206
8.11.2.1 General	206
8.11.2.2 OJPTripChangeDeliveryStructure	206
8.11.2.3 TripChangeProblemType	206
8.12 Service availability of a journey or vehicle	207
8.12.1 Description	207
8.12.2 Simple types	207
8.12.3 Request structures	208
8.12.3.1 General	208
8.12.3.2 OJPAvailabilityRequestStructure	208
8.12.3.3 BookingPTLegStructure	208
8.12.3.4 BookingUserStructure	210
8.12.3.5 ProvisionedCallAtPlaceStructure	211
8.12.4 Delivery structures	212
8.12.4.1 General	212
8.12.4.2 OJPAvailabilityDeliveryStructure	212
8.12.4.3 AvailabilityResultStructure	212
8.12.4.4 AvailabilityProblemType	213
8.12.4.5 BookableServiceItemStructure	213
8.13 Service line information	213
8.13.1 Description	213
8.13.2 Simple types	214
8.13.3 Request structures	214
8.13.3.1 General	214
8.13.3.2 OJPLineInformationRequestStructure	214
8.13.4 Delivery structures	214
8.13.4.1 General	214
8.13.4.2 OJPLineInformationDeliveryStructure	214
8.13.4.3 LineResultStructure	214
8.13.4.4 LineInformationProblemType	215
8.14 Service status of the server	215
8.14.1 Description	215
8.14.2 Simple types	215
8.14.3 Request structures	215

8.14.3.1	General	215
8.14.3.2	OJPStatusRequestStructure	215
8.14.4	Delivery structures	216
8.14.4.1	General	216
8.14.4.2	OJPStatusDeliveryStructure	216
8.14.4.3	OJPStatusResponseStructure	216
8.14.4.4	StatusProblemType	217
8.15	Discovery and Capabilities	217
Annex A (informative) A distributed approach to journey planning across Europe		218
A.1	Fares and Ticketing	221
A.2	Enriching in distributed systems	221
Annex B (informative) Lessons from experiences of Distributed Journey Planning to date		223
B.1	Introduction	223
B.2	EU-Spirit	223
B.3	Shared journey planning methodology	224
B.4	Shift2Rail	225
B.5	LinkingAlps	227
B.6	OJP4Danube	228
B.7	The way distributed journey planning is done in the Nordic countries	229
Annex C (informative) Relation between OJP and Transmodel		231
C.1	Overview	231
C.2	Relevant Diagrams from Transmodel: TRIP	231
C.3	Relevant Diagrams from Transmodel: FARE	239
C.4	Relevant Diagrams from Transmodel: SERVICE JOURNEY	242
C.5	Relevant Diagrams from Transmodel: SITUATION	243
C.6	Relevant Diagrams from Transmodel: LOCATION	245
C.7	Relevant Diagrams from Transmodel: SCHEDULE	247
Annex D (informative) Changes from OJP 1.0		248
D.1	List of Changes	248
Bibliography		254

European foreword

This document (CEN/TS 17118:2024) has been prepared by Technical Committee CEN/TC 278 “Intelligent transport systems”, the secretariat of which is held by NEN.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CEN shall not be held responsible for identifying any or all such patent rights.

This document supersedes CEN/TS 17118:2017.

Any feedback and questions on this document should be directed to the users’ national standards body. A complete listing of these bodies can be found on the CEN website.

According to the CEN/CENELEC Internal Regulations, the national standards organisations of the following countries are bound to announce this Technical Specification: Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Republic of North Macedonia, Romania, Serbia, Slovakia, Slovenia, Spain, Sweden, Switzerland, Türkiye and the United Kingdom.

0 Introduction

0.1 General

The availability of accurate and timely information about public transport (PT) services has been an increasing expectation over the past decade or more and systems have been developed to assist in the compilation and delivery of such information making best use of the rapid advances in information technology (IT) capabilities. Multi-modal information systems typically have started with an urban or regional focus to meet the information needs of those making relatively local journeys – whilst information requirements for longer-distance journeys have been delivered primarily by mono-modal information systems from the rail and airline industries.

Several existing systems employed different architectures that collated data from multiple sources to be able to offer information about longer-distance journeys that involved travel in more than one regional information system in a single transaction. This technique of bringing information together from two or more information systems when necessary is referred to as "distributed journey planning".

The ability to extend such systems to wider applications also in the context of EU/1926/2017 has been enhanced through the way in which public transport data is now increasingly standardised by following the principles set out in the "Public Transport Reference Data Model" (Transmodel) EN standard, and its related implementation standards and specifications. Terms within this document that directly relate to Transmodel are spelled in upper case letters (e.g., SERVICE JOURNEY).

0.2 An Open API for distributed journey planning (OJP)

Distributed Journey Planning systems have been in operation in Europe over the past 15 years. The observation that, whilst the architecture of each of these systems was different, the nature of the enquiries sent between the systems, and the content of the responses sent in return, were essentially the same, lead to the definition of the Open Journey Planning API to support all distributed journey planning systems. With the extension of alternative modes in Transmodel new use cases were defined which are addressed in this version.

The Open API for distributed Journey Planning (OJP) allows a system to engineer just one interface (to authorised users or openly as they so choose) rather than having to engineer separate APIs for each bipartite exchange arrangement that may be required with other systems.

0.3 The public transport information tensions

The greatest use of public transport (in terms of the number of passenger journeys) happens in urban areas where frequent and regular services cater for the needs of relatively short-distance journeys. Usage then declines as journey distances get longer – with inter-regional and international journeys comprising the smallest number of public transport journeys.

However, the need for information about public transport (PT) services is least in areas with frequent and regular services, where passengers quickly get to know about the services they rely on for most of their journeys – and other information than the schedule become more relevant in the passenger information (such as disturbances and how to move around them). Longer distance journeys, however, are made less often and for a variety of reasons there is a much greater need to obtain information for such journeys before setting off. So, the need for information is greatest for the very journeys that are made least often. Therefore, it is difficult to make a business case to provide information systems geared specifically to the needs of the longer-distance travellers. Instead, it becomes important to find ways of meeting the information needs of those passengers by using information collated and delivered primarily for the much larger group of those making short-distance journeys.

The distributed journey planning systems so far were a measured response to these tensions, allowing data to be shared across multiple systems in different ways to ensure that someone wanting to plan a journey anywhere in (for instance) Germany could go to their own regional journey planner which

collected relevant information from other regional planners in order to satisfy a particular enquiry. In Great Britain Transport Direct provided a national journey planning service collating data from the 11 regional traveline journey planners. And in the Baltic area EU-Spirit collated information from several sources to allow international planning between several European states. In Slovenia a proposal to extend the national journey planning system to a multi-national distributed system was included in the SIJPRIS service, although to date it has not yet been implemented.

Over time technology and techniques have allowed these systems to drive greater efficiencies as regional systems have been able to merge into ones covering larger areas. This has reduced the complexity of the distributed journey planning process within the areas covered by these systems. The moves to open data have also enabled larger "consolidated" datasets to be used in journey planners (notably Google Transit, or traveline's national journey planner in Great Britain) – but these are systems offering much less rich information than that from the more local or regional systems, but these systems are, due to the nature of their data sources, offering much less rich information than that from the more local or regional systems. In addition, the integration time for larger providers give operational disadvantages compared to local and regional systems, even when the same source data is available.

Taking all these factors into account there remain advantages in the distributed journey planning model notwithstanding the trends towards travel information systems that are designed to cover ever larger areas. Key points would appear to be that distributed systems are sharing the most up to date information from the local authorised source in a way that cannot be achieved with systems that collate data for much larger areas. This is particularly important in areas where local public transport market is deregulated (as has been the case in most of Great Britain since 1986) where bus services can change on any date of an operator's choosing rather than having only one or two service change dates each year from which any changes of services are known well in advance (as is the case in many other parts of Europe).

In the foreseeable future distributed journey planning will continue to provide an effective mechanism for extending the geographical scope of any journey planner with a minimum of effort – so long as there is a single standardised API as proposed in this document which will ensure that only one API would need to be engineered to allow standard questions to be asked of other systems, and to allow standard answers to standard questions from other systems to be sent in response.

For the enquiring end user there is one other important advantage of distributed journey planning – and that is that the questions can be asked, and answers read, within a layout that the user is familiar with – and in their own native language.

0.4 Distributed journey planning architecture beyond scope

0.4.1 General

Distributed Journey Planning depends not just on the availability of an API for the exchange of data. It also requires the system responding to an end-user's enquiry to be able to work out what enquiry to send to one or more other information systems, and how to merge the responses with data from its own repositories in order to create one or more seamless journey plans for the enquirer. There are several different approaches to the "architecture" for distributed journey planning – and these are beyond the scope of this document. The following paragraphs, however, outline some of the key considerations that any implementation of distributed journey planning will need to consider.

0.4.2 The distributed journey planning approach

One of the key considerations for building a distributed journey planning system is to define what supporting data (metadata) is required and where it is to be held. At its simplest the process of making an enquiry typically has several stages (see also Figure 9):

- a) An enquirer goes to his/her **home system** and composes an enquiry expressing the location of the start and end points in their own terms or as permitted by the user interface.

- b) The enquirer's **home system** seeks to match the enquirer's locations to locations understood by the journey planner, expressed in terms (perhaps geographic coordinates) that can be understood by the home and other distributed journey planning engines.
- c) The home system establishes what questions it needs to ask and which journey planning systems (both its own and those of one or more distributed partners) may be able to provide the information needed.
- d) The **home system** then collates the information received from the different systems to create a seamless and efficient journey plan which it can then deliver to the enquirer.

In some systems the **home system** does not itself undertake the distributed journey planning. Instead the **home system** passes that task to a separate distributing journey planning system which completes the process and returns the answers to the **home system**.

For the enquirer it is important to make the process as simple and efficient as possible – so the process of matching locations with system gazetteers can be a critical one. Ideally the enquirer should be able to specify a location as a station or stop name, a topographic place, a street address, a postcode (if this covers a meaningful small area), and possibly Points of Interest. Such data for locations within the geographical scope of the **home system** is likely to be held already – but if the location is outside that geographical scope, where does the equivalent data come from? And how does the home system know that it needs to find data for a distant location?

0.4.3 Distributed or centralised approaches

One of the key considerations for building a distributed journey planning system is to define what supporting data (metadata) is required and where it is to be held. Somehow the home system needs to be able to recognise that a requested origin or destination is not in its own geographical area – and once it has done that it also needs to recognise which system(s) will be able to provide journey planning answers for that location. One way of managing this would be for a network of distributed journey planning systems to share a central repository of gazetteers (indices of geographical entities – localities, addresses, stops & stations, etc.) to resolve these questions (and probably to go on to make the necessary enquiries to the relevant journey planning systems) before handing back the information to the originating home system. This would be a centralised model for handling journey enquiries. Alternatively, each participating system could hold gazetteer data for all the participating systems' areas – and the enquiry process could then work on a peer-to-peer decentralised basis. To get an efficient approach it is necessary to consider not only the gazetteers, but also how to access the timetable data for long-distance PT services (notably trains, coaches, ferries, and flights).

0.4.4 The basis for the Open API

Work in Germany's IP-KOM research project has brought together the lessons learned from various information systems (including EU-Spirit, JourneyWeb and DELFI) and developed the TRIAS¹ schema to support future information systems in Germany. From TRIAS the Open API for distributed journey planning was evolved.

Several existing European Standards and Technical Specifications will underpin the work – notably Transmodel (Public Transport Reference Data Model), SIRI (Service Interface for Realtime Information) and NeTEx (Network and Timetable Exchange). The Open API will depend on the consistency of the data from all sources that will become available by the implementation of the existing European Standards for public transport information.

¹<https://knowhow.vdv.de/documents/431-1/>
<https://knowhow.vdv.de/documents/431-2/>

0.4.5 Other possible uses for the Open API

Whilst the Open API is intended primarily to support distributed journey planning, experience of such APIs to date has shown that they can also be used for other purposes. For instance, they can be used for communication between personal journey planning apps and a journey planning service (without any distributed journey planning requirement). Or they can be used to enable a park-and-ride planner to combine car journey planning with public transport journey planning, connecting the two modes at the park-and-ride car parks. Or they can support the use of taxis as a mode to access public transport in areas where conventional public transport does not exist or is very sparse. A standard Open API will provide many opportunities to use and re-use public transport and associated data in the delivery of innovative information services.

A seamless integration on the process level with distribution API is also considered important especially in a multi-modal or inter-modal use case. The planned trips should feed in a consistent way into such API.

Currently, no CEN standard for distribution (booking, payment, ticketing) is considered. We considered two relevant API specifications for distribution to define the scope of OJP and to support the process as defined in Figure 1:

- TOMP (<https://github.com/TOMP-WG/TOMP-API>): A standard mainly for MaaS purposes.
- OSDM (<https://osdm.io/>): A new standard favoured by UIC for international railway for distributed booking.

The idea is to support a seamless operation, so that a home system can use information from OJP services to subsequently do reservations/bookings for its users.

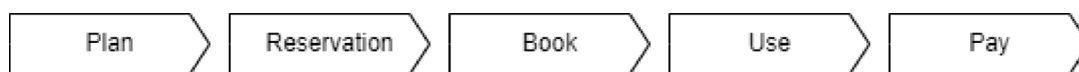


Figure 1 — Processes around travel

0.5 The European ITS Directive

The Delegated Regulation of the ITS Directive (EU/1926/2017)² regarding the provision of EU-wide multimodal travel information services (MMTIS) will provide the necessary requirements to make EU-wide MMTIS accurate and available across borders. It establishes the specifications necessary to ensure the accessibility, exchange and update of travel and traffic data and distributed journey planning for the provision of multimodal information services in the European Union. The Delegated Regulation recommends the use of the Open API for distributed journey planning standard in [16] and in relation to the requirements specified in Article 7 'Linking Travel information Services'.

² <https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32017R1926>

1 Scope

This document defines a schema for establishing an Open API for Distributed Journey Planning that can be implemented by any local, regional, or national journey planning system in order to exchange journey planning information with any other participating local, regional or national journey planning system.

NOTE Especially for National Access Points under EU/1926/2017, this API is relevant.

2 Normative references

There are no normative references in this document.

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

NOTE 1 This section provides the list of terms used in the OJP schema with definitions based on the definitions of the CEN standards Transmodel, NeTEx, SIRI.

NOTE 2 The dictionary definitions of words in everyday use, such as Journey and Trip, are not sufficiently precise to be used in the specification of an information system – and therefore these (and other) words have very specific definitions (as shown below) when used in the technical sections of this document (sections 3 onwards). A Journey Planning System can have many outputs – in the context of this document the focus is on its Trip Planning functionality.

3.1 Terms used in OJP schema

NOTE 1 the correspondence to the Transmodel, NeTEx and SIRI terms is found in https://github.com/VDVde/OJP/tree/master/docs/transmodel_ojp_mapping.

NOTE 2 TMv6.2 references the discussions that will result in a revision in 2023 or 2024. We referenced those as they are adapted for the better alignment of Transmodel and OJP.

3.1.1

AbsoluteBearing

absolute compass bearing in degrees

3.1.2

AccessFeatureType

[corresponds to AccessFeatureType of PATH LINK in TMv6]

type of physical feature of PATH LINK

3.1.3

AccessibilityFeature

[corresponds to AccessFeatureType in SIRI]

property of PATH LINK of interest to visually, auditorily, or mobility impaired passengers

3.1.4

Address

[TMv6]

descriptive data associated with a PLACE that can be used to describe the unique geographical context of a PLACE for the purposes of identifying it. May be refined as either a ROAD ADDRESS, a POSTAL ADDRESS or both

3.1.5

Altitude

altitude in metres above sea level

3.1.6

Attribute

generic attribute (with associated text) that can be associated with many facilities, PT service restrictions, etc.

3.1.7

BookingArrangement

[generalisation of BOOKING ARRANGEMENTS in TMv6]

arrangement for booking any TRIPLE(s)

3.1.8

CalcTime

calculation time in milliseconds

3.1.9

CallAtNearStop

indication of the walk distance and time to a nearby stop where relevant

3.1.10

CallAtStop

[same as Call in SIRI]

meeting of a VEHICLE JOURNEY with a specific SCHEDULED STOP POINT

3.1.11

CertificateId

form of identification that can be used as a Message Integrity Property (public key cryptography)

3.1.12

Connection

[corresponds to CONNECTION in TMv6.2]

physical (spatial) possibility for a passenger to change from one public transport vehicle to another to continue the trip, determined by a pair of points which may be either SCHEDULED STOP POINTs or VEHICLE MEETING POINTs

Note 1 to entry: Different times may be necessary to cover the link between these points, depending on the kind of passenger.

3.1.13

ContinuousLeg

[a specialisation of RIDE LEG in with Timed=false, a PERSONAL LEG or an ACCESS LEG TMv6 and NeTEx]

LEG of a TRIP that is not bound to a timetable. It can either be a RIDE LEG with Timed=false, a PERSONAL LEG or an ACCESS LEG

3.1.14

ContinuousModes

modes that run at any time without a timetable

3.1.15

ContinuousService

[specialisation of SERVICE JOURNEY in TMv6]

vehicle movement on a continuous, non-timetabled service

3.1.16

DataFrameRef

[corresponds to VERSION FRAME of data in TMv6]

identifier of the set of data being used by an information system, which allows a comparison to be made with the versions of data being used by overlapping systems. Corresponds to VERSION FRAME Id

3.1.17

DatedJourney

[corresponds to a specialisation of a DATED VEHICLE JOURNEY in TMv6]

passenger carrying VEHICLE JOURNEY for one specified DAY TYPE for which the pattern of working is in principle defined by a SERVICE JOURNEY PATTERN. DatedJourney details include its operating days

3.1.18

Delivery

process of delivering messages

3.1.19

DirectionCode

[related to DIRECTION in TMv6]

identifier of a group of DIRECTIONS of the ROUTEs belonging to the same LINE created for the purpose of filtering and organising timetables

3.1.20

Distance

physical measurement of the distance between two specific points in metres

3.1.21

EntitlementProduct

[corresponds to ENTITLEMENT PRODUCT in TMv6 and in NeTEx that may be materialised as a TRAVEL DOCUMENT]

precondition to access a service or to purchase a FARE PRODUCT issued by an organisation that may not be a PT operator (e.g.: military card, concessionary card, etc)

3.1.22

Equipment

[TMv6]

item of equipment installed either fixed (PLACE EQUIPMENT) or on-board vehicles (VEHICLE EQUIPMENT). A service (LOCAL SERVICE such as LEFT LUGGAGE, TICKETING SERVICE) is considered as immaterial equipment as well. TYPE OF EQUIPMENT is a classification of EQUIPMENT

3.1.23

ExchangePoints

[specialisation of CONNECTION END in TMv6]

points at which partial solutions from two different journey planning systems may establish a connection

3.1.24

Facility

[corresponding to FACILITY in TMv6]

named amenity available to the public at a SITE or on a SERVICE. A FACILITY has no further properties other than a name

Note 1 to entry: An EQUIPMENT or LOCAL SERVICE is used to describe the further properties provided as part of particular FACILITY

3.1.25

Fare

[corresponds to FARE PRODUCT in TMv6 and NeTex]

immaterial marketable element (access rights, discount rights, etc.), specific to a CHARGING MOMENT.

3.1.26

FareAuthority

[view of the aggregation of ORGANISATION and RESPONSIBILITY ROLE in TMv6]

ORGANISATION which has the RESPONSIBILITY ROLE for the definition of FARE PRODUCTS

3.1.27

FareParam

parameters which are provided in the Fare Price Models of Transmodel/NeTex

3.1.28

FarePassenger

[specialisation of USER PROFILE in TMv6 and NeTex]

attributes of a passenger that influence the price to be paid by that passenger for a FARE PRODUCT

3.1.29

FareProductBooking

[aggregation of DISTRIBUTION CHANNEL and BOOKING ARRANGEMENT in TMv6 and NeTex]

information to support the process of booking a FARE PRODUCT

3.1.30

FareProductPrice

[corresponds to FARE PRODUCT PRICE in TMv6 and NeTex]

price at which a FARE PRODUCT can be purchased

3.1.31

FareProductValidity

[TMv6 and NeTex]

conditions of use for a particular FARE PRODUCT

3.1.32

GeoArea

[specialisation of ZONE in TMv6]

specialisation of a ZONE defined by a two-dimensional PLACE within the service area of a public transport operator (administrative zone, TARIFF ZONE, ACCESS ZONE, etc.)

3.1.33

GeoCircle

definition of a circular GeoArea

3.1.34

GeoPosition

[generalisation of LOCATION in TMv6]

Longitude, Latitude in WGS84 (and Altitude if required)

3.1.35

GeoRectangle

definition of a rectangular GeoArea

3.1.36

GeoRestriction

application of geographical restrictions to a search

3.1.37

GuidanceAdvice

various types of guidance advice given to travellers

3.1.38

IndividualMode

mode which an individual powers themselves (such as walk, cycle)

3.1.39

IndividualTransportOption

individual mode and its usage limits as stated by the passenger (TRIP ACCESS CONSTRAINT)

3.1.40

InitialLocationInput

input of a GeoPosition for a starting point

3.1.41

InternationalText

[specialised view of ALTERNATIVE NAME in TMv6]

alternative identified text to be used in specified languages

3.1.42

Journey

[specialisation of JOURNEY in TMv6]

corresponds to a NORMAL DATED VEHICLE JOURNEY or to a DATED VEHICLE JOURNEY but may be extrapolated from a VEHICLE JOURNEY

3.1.43

Latitude

latitude in degrees from equator

3.1.44

Leg

[LEG according to TMv6.2 covering more MODEs]

single stage of a TRIP that is made without change of MODE or VEHICLE JOURNEY (e.g., between each TRANSFER).

3.1.45

LegAlight

situation at a stop or station at which the passenger alights from a LEG including time-related information

3.1.46

LegAttribute

facilities available on a specified (part of a) LEG

3.1.47

LegBoard

situation at a stop or station at which the passenger boards a LEG including time-related information

3.1.48

LegIntermediate

situation at a stop or station that lies between the LegBoard and LegAlight stop or station including time-related information

3.1.49

LegTrack

[corresponds to LEG TRACK in TMv6]

LINK PROJECTION of a LEG onto the topography of the route being followed

3.1.50

Length

distance in metres between two points normally along a defined route

3.1.51

Line

[corresponds to LINE in TMv6]

group of ROUTEs which is generally known to the public by a similar name or number

3.1.52

LineDirection

[corresponds to DIRECTION in TMv6]

GROUP of DIRECTIONS of the ROUTEs belonging to the same LINE created for the purpose of filtering and organising timetables

3.1.53

LineIdentity

[attribute of LINE in TMv6]

number, letter or name that identifies a particular LINE

3.1.54

LinkProjection

[corresponds to LINK PROJECTION in TMv6]

oriented correspondence from one LINK of a source layer, onto an entity in a target layer: e.g., LINK SEQUENCE, COMPLEX FEATURE, within a defined TYPE OF PROJECTION.

3.1.55

Location

spatially-related input (such as name, postcode, coordinate, etc) given by an enquirer and used by the LocationInformation service

3.1.56

LocationInformation

information about a specific PLACE or LOCATION

3.1.57

LocationInformationService

service establishing the existence in an information system of a location of various types as seen by an enquirer

3.1.58

LocationText

text used by an enquirer of an information system to indicate an origin, destination or other location

3.1.59

Longitude

longitude in degrees from Greenwich

3.1.60

MessageIntegrityProperties

properties that allow checks for message integrity

3.1.61

Mode

[specialisation of MODE in TMv6]

extended range of VEHICLE MODEs, aggregating them with some SUBMODEs

3.1.62

MultiPointTrip

[special concept relevant to distributed journey planning]

Trip that is defined by multiple alternative Origins and/or Destinations

3.1.63

MultiTripFare

[specialisation of FARE PRODUCT in TMv6 and NeTEx]

one or more FARE PRODUCTS covering multiple TRIPs or LEGs

3.1.64

NoChangeAt

no-change-at restrictions for a TRIP, e.g., SCHEDULED STOP POINTs or STOP PLACEs at which no TRANSFER is allowed within a TRIP

3.1.65

NotVia

[related to TRIP VIA PLACE in TMv6]

not-via restrictions for a TRIP, e.g., SCHEDULED STOP POINTs or STOP PLACEs that the TRIP is not allowed to pass through

3.1.66

NumberOfResults

parameter to control the number of TRIP results before/after a point in time. May NOT be used when departure time at origin AND arrival time at destination are set

3.1.67

Occupancy

[corresponds to Occupancy in SIRI]

passenger load status of a VEHICLE

3.1.68

OJPService

IT service within the OJP schema

3.1.69

OJPSubscription

subscription arrangement for a participant in a Distributed Journey Planning system

3.1.70

OpenPercent

percent value as integer, no upper limit

3.1.71

OperatingDay

[corresponds to OPERATING DAY in TMv6]

day of public transport operation of which the characteristics are defined in a specific SERVICE CALENDAR and which may last more than 24 hours

3.1.72

Operator

[corresponds to OPERATOR in TMv6]

company providing public transport services

3.1.73

OptimisationMethod

type of algorithm that can be used for planning a TRIP (fastest, least walking, etc)

3.1.74

OsmTag

Key-value tag used in the OpenStreetMap system

3.1.75

Owner

[specialisation of an ORGANISATION in TMv6]

ORGANISATION with ownership as the RESPONSIBILITY ROLE

3.1.76

Participant

[corresponds to Participant in SIRI]

IT system that is participating in a communication with other participant(s)

3.1.77

PassengerCategory

[specialised simplified view of USER PROFILE in TMv6 and NeTEx]

classification of passengers by age or other factors that may determine the fare they will need to pay

3.1.78

PathGuidance

[extended view of a NAVIGATION PATH in TMv6 to include the textual navigation instructions, refers to PATH GUIDANCE (an instruction to a passenger for an individual step when following a path between the start and the end of a TRANSFER LEG) in TMV6]

description of a piece of a TRIP. May include geographic information, turn instructions and accessibility information

3.1.79

PathGuidanceSection

one or more path guidance sections that build the LEG of a TRIP

3.1.80

PathLink

[specialised view of a PATH LINK in TMv6]

link within a PLACE or between two PLACES (that is STOP PLACES, ACCESS SPACES or QUAYS, BOARDING POSITIONS, POINTS OF INTEREST etc or PATH JUNCTIONS) that represents a step in a possible route for pedestrians, cyclists or other out-of-vehicle passengers within or between a PLACE

3.1.81

PERSONAL LEG

[specialised view of a PATH LINK in TMv6]

part of a TRIP PATTERN corresponding to the movement of a user (passenger, driver) from one PLACE to another, where the trip is made entirely by personal mode and neither of the endpoints is a SCHEDULED STOP POINT or VEHICLE MEETING POINT. This will be modelled as a ContinuousLeg.

3.1.82

Percent

percent value as integer, limited to max value 100

3.1.83

PersonalMode

[category of MODE in TMv6, refers to PERSONAL MODE]

MODEs offered by private individuals

3.1.84

PhoneNumber

[specific attribute of CONTACT DETAILS in TMv6]

phone number for a specified organisation, department or purpose

3.1.85

Place

[corresponds to PLACE in TMv6]

geographic PLACE of any type which may be specified as the origin or destination of a trip

3.1.86

PlaceContext

[view of PLACE in TMv6]

PLACE and access to it by individual transport

3.1.87

PlaceData

types of PLACE-related data that can be used as filters

3.1.88

PlaceType

[corresponds to TYPE OF PLACE in TMv6]

classification of basic types of PLACE

3.1.89

PlaceUsage

use of a PLACE as an origin, destination or via

3.1.90

PointOfInterest

[corresponds to POINT OF INTEREST in TMv6 with related information]

type of PLACE to or through which passengers may wish to navigate as part of their TRIP and which is modelled in detail by journey planners

3.1.91

PointOfInterestCategory

[view of POINT OF INTEREST CLASSIFICATION in TMv6]

categorisation of POINTs OF INTEREST in respect of the activities undertaken at them

3.1.92

Priority

priority values ranging from 1 (highest) to 5 (lowest)

3.1.93

PrivateCode

code within scope of a private referential system

3.1.94

PrivateService

specific service operated by a Private Mode

3.1.95

Problem

message structure that signals problems or errors

3.1.96

ProgressBetweenStops

[similar to SIRI]

type for progress between stops

3.1.97

PtMode

[from SIRI]

mode of public transport service, corresponds to VEHICLE MODE and SUBMODE

3.1.98

PtSubmode

[corresponds to SUBMODE in TMv6]

submode of a public transport mode

3.1.99

RouteDescription

[attribute of ROUTE in TMv6]

descriptive text for a ROUTE, e.g.: "Airport via City Centre"

3.1.100

Service

context-dependent offering of a given facility

Note 1 to entry: Such facilities include:

- PT service of an overall package of one or more public transport routes, timetables and travel facilities offered;
- IT service;
- journey planning, information, or other functional service; or
- an immaterial facility such as Left Luggage

Note 2 to entry: The context is often clear when it is used with another term (as in PT service, special service, service link, service pattern, etc)

3.1.101

ServiceDelivery

[IT service, taken from SIRI]

response from producer to consumer to deliver payload data

Note 1 to entry: Either answers a direct ServiceRequest or asynchronously satisfies a subscription.

3.1.102

ServiceFacility

[view of a SERVICE FACILITY SET in TMv6]

(view of) SERVICE FACILITY SET (a set of FACILITIES available for a specific VEHICLE TYPE (e.g., carriage equipped with low floor) possibly only for a service (or for a SERVICE JOURNEY or a JOURNEY)

3.1.103

ServiceRequest

[an IT service, taken from SIRI]

request for the immediate delivery of data. Answered with a ServiceDelivery (or a DataReadyRequest)

3.1.104

ServiceResponse

[an IT service, taken from SIRI]

response message containing a payload and various attributes of its own

3.1.105

ServiceStatus

parameters which describe the current status of a DATED VEHICLE JOURNEY

3.1.106

ServiceViaPoint

[specialisation of VIA in TMv6]

STOP PLACE or SCHEDULED STOP POINT as a VIA for a particular SERVICE PATTERN. Specialisation of a VIA

3.1.107

SharingService

transport service that provides shared vehicles

3.1.108

Signature

data for transmission of message signatures (public key cryptography), used to prove Message Integrity

3.1.109

SituationFullRef

reference to situation message. Message details might be found in response context or through other communication channels

3.1.110

Situations

[as SIRI]

container for the structured description of situations in public transport or on the road (individual transport)

3.1.111

Speed

relative speed in percent. If given slows the standard speed (below 100) or increases it (above 100)

3.1.112

StaticFare

[specialisation of a FARE PRODUCT in TMv6 and NeTEx]

list of FARE PRODUCTS under the responsibility of a FARE AUTHORITY, independent of any specific trip result

3.1.113

Status

whether the request was processed successfully or not

Note 1 to entry: Default is "true"

3.1.114

StopAttributes

[properties of a SCHEDULED STOP POINT or STOP PLACE in TMv6]

selection of attributes of ACCESSIBILITY LIMITATION related to a SCHEDULED STOP POINT extended by attributes referring to some PLACE EQUIPMENT

3.1.115

StopCallStatus

[see SIRI CALL]

current status of each stop along the route of a PT Service

3.1.116

StopEvent

[see SIRI CALL]

departure or arrival event or both

3.1.117

StopEventContent

[see SIRI CALL]

parameters which define the extent of information to be returned in a Stop Event result

3.1.118

StopEventData

[see SIRI CALL]

parameters which establish various filtering criteria for Stop Event results

3.1.119

StopEventParam

[see SIRI CALL]

parameters which define what is to be included in a Stop Event result

3.1.120

StopEventPolicy

[see SIRI CALL]

parameters which define the number, Time Window and types of Stop Event to be included

3.1.121

StopFacility

[corresponds to a SITE FACILITY SET in TMv6]

FACILITY that applies to a stop

3.1.122

StopFare

stop-related FARE PRODUCT information

3.1.123

StopPlace

[extended view of STOP PLACE in TMv6]

STOP PLACE extended by ACCESSIBILITY LIMITATION properties and of the associated equipment, comprising one or more locations where vehicles may stop and where passengers may board or leave vehicles or prepare their trip, and which will usually have one or more well-known names

3.1.124

StopPoint

[extended view of SCHEDULED STOP POINT in TMv6]

SCHEDULED STOP POINT extended by ACCESSIBILITY LIMITATION attributes and with identifier and name where passengers can board or alight from vehicles

3.1.125

Submode

[corresponds to SUBMODE in TMv6]

variant of a MODE, as for instance international or domestic rail (rail being the MODE)

3.1.126

Subscription

[see SIRI]

arrangement which allows a specific organisation to access an information system - usually taking the form of security parameters which may or may not need to be paid for

3.1.127

SubscriptionRequest

[see SIRI]

request from a subscriber to a producer for a subscription to a particular IT service. Answered with a SubscriptionResponse

3.1.128

TariffZone

[Corresponds to TARIFF ZONE to TMv6 and TariffZone in NeTEx]

ZONE used to define a zonal fare structure in a zone-counting or zone-matrix system

3.1.129

TerminateSubscription

[see SIRI]

request to end a subscription to a particular IT service

3.1.130

Ticket

[corresponds to a FARE PRODUCT in TMv6 and NeTEx]

FARE PRODUCT which may be materialised as a TRAVEL DOCUMENT and which gives the passenger an entitlement to travel on one or more specific LEGs subject to specified conditions

3.1.131

TimedLeg

[corresponds to a RIDE LEG with the attribute Timed in TMv6.2 (with related information)]

passenger LEG with timetabled schedule

3.1.132

TimeWindow

window of opportunity that the traveller has to perform a specific LEG

3.1.133

TopographicPlace

[corresponds to TOPOGRAPHIC PLACE in TMv6]

type of PLACE providing the topographical context when searching for or presenting travel information, for example as the origin or destination of a trip

Note 1 to entry: It may be of any size (e.g., County, City, Town, Village) and of different specificity (e.g., Greater London, London, West End, Westminster, St James's)

3.1.134

TrackSection

LINK PROJECTION on the infrastructure network of the trip leg section together with time information.

3.1.135

Transfer

[corresponds to TRANSFER in TMv6 and Transfer in NeTEx]

travel between two POINTs located sufficiently near to each other that they represent for a passenger a possibility to reach one of the POINTs when starting at the other in a timescale which is realistic when undertaking a TRIP, e.g.: ACCESS or CONNECTION

3.1.136

TransferDuration

[attribute of a CONNECTION (not INTERCHANGE) in TMv6]

calculated duration in a response taking into account the request parameters.

Note 1 to entry: TransferDuration plus waiting time is the minimum interval between arrival and departure time

3.1.137

TransferLeg

[specialisation of TRANSFER LEG in particular CONNECTION LEG in TMv6.2]

description of a LEG which links other LEGs where a TRANSFER or CONNECTION between different LOCATIONs is required

3.1.138

TransferModes

MODEs dedicated to perform TRANSFERs

3.1.139

Transition

transition types for interchanges

3.1.140

Trip

[extended view of TRIP PATTERN according to TMv6.2 (to include time-related information)]

whole journey from passenger origin to passenger destination in one or more LEGs

3.1.141

TripContent

parameters that control the level of detail of the TRIP results

3.1.142

TripData

data to be included/excluded from search, e.g., modes, operators, etc

3.1.143

TripFare

Fare for a specified TRIP

3.1.144

TripFareProduct

[specialisation of FARE PRODUCT in TMv6 and NeTEx]

FARE PRODUCT covering a part or the whole of a TRIP from boarding the first public transport vehicle to alighting from the last public transport vehicle (corresponds to a package of PREASSIGNED FARE PRODUCTS)

3.1.145

TripInfoService

short for Trip Information Service. OJP Service that provides details on a single VEHICLE JOURNEY that may be used by a passenger TRIP

3.1.146

Leg

[corresponds to LEG according to TMv6.2 covering more MODEs]

single stage of a TRIP that is made without change of MODE or VEHICLE JOURNEY (ie: between each TRANSFER)

3.1.147

TripLocation

location of a passenger currently travelling in a VEHICLE

3.1.148

TripMobility

parameters the user can set to restrict the mobility options, particularly for interchanging

3.1.149

TripParam

parameters which define the content expected in a TRIP plan

3.1.150

TripPolicy

parameters which control the TRIP search behaviour

3.1.151

TripSummary

summary of the details of a TRIP

3.1.152

TripVia

[relates to TRIP VIA PLACE in TMv6]

VIA restrictions for a TRIP

3.1.153

TurnAction

range of alternative turns that can be described

3.1.154

TypeOfFareClass

classes of travel available on a particular PT service which will affect the price to be paid

3.1.155

ValidDayBitType

sequence of bits (values 0 or 1) coded as a text string and representing days

3.1.156

VatRate

relevant Value Added Tax rate

3.1.157

Vehicle

[a specialisation of VEHICLE in TMv6 to include privately operated vehicles as well as public transport ones]

specific type of vehicle used for carrying passengers

3.1.158

VehiclePosition

geographical and logical position of a vehicle

3.1.159

VehicleProgress

progress of a vehicle relative to timetable service pattern

3.1.160

Version

[corresponds to VERSION in TMv6]

group of operational data instances which share the same VALIDITY CONDITIONs. A version belongs to a unique VERSION FRAME and is characterised by a unique TYPE OF VERSION

3.1.161

WebLink

URL of a web resource with label

3.1.162

Weekday

[derived from DAY OF WEEK and PROPERTY OF DAY in TMv6]

enumeration of individual the seven DAYS OF WEEK, along with public holidays

3.1.163

WeekdayTimePeriod

[specialisation of TIME BAND in TMv6]

aggregation of TIME BAND, DAY OF WEEK and PROPERTY OF DAY

3.1.164

Zone

[corresponds to ZONE in TMv6]

two-dimensional PLACE within the service area of a public transport operator (administrative zone, TARIFF ZONE, ACCESS ZONE, etc.)

3.2 Terms in OJP Schema

The following are extensions that can be associated with other Dictionary entries

3.2.1

Base

basic or default requirement

3.2.2

Code

shorthand reference for a particular item of data

3.2.3

Enumeration

set of values that can be associated with the term

3.2.4

Filter

parameter that controls the level of detail of a particular type of information

3.2.5

FilterGroup

collection of parameters that control the level of detail of a particular type of information

3.2.6

FilterStructure

structure of a set of filter parameter(s)

3.2.7

Group

group of elements related to a particular topic

3.2.8

Input

requirement for specific type of information to be supplied for the system to work

3.2.9

InputStructure

structure of the way an INPUT is to be expressed

3.2.10

Ref

code that refers to something specific, normally of a specified type (such as VEHICLE JOURNEY, STOP POINT, DAY TYPE, etc)

3.2.11

RefGroup

group of REFs related to a specific topic

3.2.12

ReferenceStructure

RefStructure

structure of the way that a REF code is expressed in a particular context

3.2.13

Request

structured message requesting specific information

3.2.14

RequestStructure

structure of a Request for a particular type of information

3.2.15

Response

structured message providing specific information that has been requested

3.2.16

ResponseStructure

structure of a Response for a particular type of information

3.2.17

Result

partial or complete outcome from a data search or calculation including status information

3.2.18

ResultStructure

structure of a Result from an enquiry

4 Symbols and abbreviations

API	Application Programming Interface – a software interface offering services to other pieces of software
CEN	European Committee for Standardization
CENELEC	European Committee for Electrotechnical Standardization
DJP	Distributed Journey Planning
DRT	Demand-Responsive Transport – a type of bookable (public) transport where the serviced stops or departure times adapt to user demands
DS	Distributing System
EHS	Enquirer's Home System
EU	European Union
EN	European Norm
HTTP	HyperText Transfer Protocol
IFOPT	Identification of Fixed Objects in Public Transport
IP-KOM	Internet-Protokoll basierte Kommunikationsdienste (IP-based communications services)
ISO	International Standards Organisation
IT	Information Technology
ITS	Intelligent Transport Systems
LIR	LocationInformationRequest service of OJP
MaaS	Mobility as a Service
MPTR	MultiPointTrip service of OJP

NEN	Royal Netherlands Standardization Institute
NeTEx	Network and Timetable Exchange
OJP	Open API for distributed Journey Planning
Open API	An API which is published for widespread implementation
OSDM	Open Sales and Distribution Model
POI	Point of Interest
PT	Public Transport
REST	Representational State Transfer
RS	Responding System
SER	StopEventRequest service of OJP
SIRI	Service Interface for Real-time Information
SOAP	Simple Object Access Protocol
TM	Transmodel
TOMP-API	An open API for technical communication between Transport Operators and Mobility-as-a-service Providers
TPEG	Transport Protocol Experts Group
TR	TripRequest service of OJP
TRIAS	Travellers' Realtime Information and Advisory Standard
TS	Technical Specification
UIC	Union Internationale des Chemins de fer / International union of railways
UML	Unified Modelling Language
URI	Uniform Resource Identifier
URL	Universal Resource Locator
VDV	Verband Deutscher Verkehrsunternehmen (D)
WGS	World Geodetic Standard
WSDL	Web Services Description Language
XML	eXtensible Mark-up Language
XSD	XML Schema Document

5 Use cases

5.1 General

The following use cases describe the requirements for the standardised OJP schema that is being proposed for Distributed Journey Planning (DJP). It covers both key tasks that are essential for the core functions in a DJP system and some additional tasks that might usefully complement such systems. It is suggested that all implementations should cover the essential tasks – and each should also describe in a discovery mechanism those other functions which that particular implementation is able to handle.

The following descriptions assume a simple and straightforward example of Distributed Journey Planning in which there are four key "actors" or "roles" in the process:

- 1) **enquirer** – the person asking for information
- 2) the enquirer's **home system** – the journey planning system to which the enquirer is connected
- 3) a **distributing system** - the system that distributes journey planning enquiries to other systems
- 4) **responding system(s)** – the system(s) that respond to questions from the distributing system.

An enquirer requests a trip from a journey planning system which is referred to as the enquirer's **home system** – and the home system then has to recognise that the enquiry requires information from other systems. It therefore passes the enquiry to a **distributing system** (which may be integrated with the home system or separated) which makes enquiries of one or more **responding system(s)** from which it will glean additional information to put together a comprehensive trip plan to meet the enquirer's request. Once the **distributing system** has the required comprehensive trip plan it returns this to the home system, which in turn presents the results to the enquirer.

The **home system** will have a user interface through which it receives enquiries from enquirers, and through which it responds to those enquirers. It may have its own journey planning capability for a specific geographical territory (which could also cover only certain modes of transport) – or it may only have sufficient intelligence to know that it needs to draw information from other journey planning systems by passing the enquiry to the distributing system.

The **distributing system** will need to know from which journey planning systems it needs to draw all relevant information to meet the requirements of each enquiry it receives, and it needs to be able to communicate with the relevant responding systems – which it would do using an API.

To describe the geographical context of a distributed journey planning system there are several concepts. One is based on regions:

- a) the **local region** – the territory for which the home or distributing system itself can plan trips without information from other systems
- b) the **adjacent regions** – the regions which are adjacent to the local region and have their own "local" journey planning systems
- c) **remote regions** – the regions which have their own "local" journey planning systems, but which are not adjacent to the local region

A responding system has a region, when it can always deliver the answers for the complete region without resolving to a different trip planner. However, many responding systems will not be area-complete:

- A trip planner for a city or commune may include all buses, tramways, and metros, but does not inform about rail even for intra city or commune trips.
- Also trip planners associated with a single operator (e.g., long distance bus) may not cover most services available within the operation area of the operator.

There are the following typical distributed journey planning scenarios (in addition to the non-distributed scenario of a local point to a local point), each of which may require handling in a different way depending on the architecture of the distributed journey planning system:

- i) local region to remote region.
- ii) local region to adjacent region.

iii) remote or adjacent region to remote or adjacent region

The other concept uses one or several long-distance systems and then several local ones. The core is the system and there is always an overlap between the areas a system can work on. For example, Flixbus has no region where it can plan to the doorstep on its own. In this concept we have:

- 1) long distance responding systems
- 2) local public transport responding systems
- 3) local intermodal responding systems
- 4) road planning systems.

In this concept the following scenarios are used:

- Search with long distance responding system and multiple local ones
- Search with a local public transport responding system and local intermodal systems and a road planning system.
- Only road planning system (possibly with a local intermodal responding system)

In those scenarios the importance of exchange points is greatly diminished. Each stop of a long-distance system is automatically an exchange point of a local system.

As noted earlier, this document is concerned with the communication protocols necessary to pass enquiries and responses between participating DJP systems. It is not concerned with the architecture or logic of the journey planning engine that sits in any distributing system, through which the user's enquiry is broken down into appropriate components. Each component is then forwarded to a relevant local journey planning system with DJP capabilities (a **responding system**) to collect information about each component of the requested trip plan.

5.2 Key tasks for Distributed Journey Planning

5.2.1 Planning of a trip

An enquirer wishes to plan a trip and places the enquiry with a journey planning system that has DJP capabilities (the enquirer's **home system**). Part of the requested trip lies in an area covered by a different journey planning system which also has DJP capabilities. The initiating **home system** (the one which received the end-user's enquiry) passes the journey planning details to a **distributing system** which creates a journey planning request to pass to the **responding system** for the component of a trip in the responding system's territory. The request comprises one or more origin points, one or more destination points, a preferred departure or arrival time at either all of the origin or all of the destination points, and any preference that the enquirer may have stipulated for modes (and sub-modes) of transport to be included in or excluded from the trip plan, along with any specific accessibility and mobility requirements. Maximum walk distances and estimated walk speeds should be included, with a separate connection walk speed that may be different. It may also include a via (or not via) point, or a point at which a transfer may or may not take place, and it could specify preferred operator(s) of PT services, or specific PT services. It could include the journey planning optimisation that is preferred (such as fastest, least walking or least connections). And it could ask for the response to just give the essential information, or it could ask for more details, such as all stops to be shown, with scheduled times or currently projected times from a real-time system. If cycling is offered as an option to reach the start of a public transport trip (or to complete the trip) then this would need to be requested and parameters would specify maximum cycle distance and expected cycling speed. Finally, the number of trip options to be returned needs to be specified – in terms of a number of trip options, or a time period to be covered by the trip options offered,

and whether trips are required from each requested origin or just any one or more of them, and likewise to each requested destination or only to any one or more of them.

5.2.2 Discovering relevant stops

In order to specify a DJP request it will be necessary for the **distributing system** to determine the set of stops from one of which the requested component of the trip has to start – and likewise the set of stops at one of which it has to end. This set of stops are those to or from which the **distributing system** is able from its own resources (or from responses to other component enquiries) to plan the requested overall trip. There may be a need, therefore, to establish the identity of potential connection stops on the border between journey planning systems. This would require an exchange of questions and answers, delivering the name of the stop, its coordinates, and the code by which it is recognised by the relevant **responding systems**.

5.2.3 Obtaining information about accessibility and services for those with special needs

Travellers with mobility, sensory or other disabilities may have special needs when using public transport. The enquiring user needs to be able to describe their own needs in a simple and codified way that can be interpreted intelligently by journey planners, so that solutions offered best meet the needs of the enquirer. The schema for DJP systems therefore has to be able to convey the specific needs of such travellers, whilst the responses have to be able to carry the information necessary to describe the accessibility or other features of the PT services being proposed in the trip plan in sufficient detail that the traveller can be confident of what will be offered when making the trip. For this to be matched to available information about accessibility restrictions it would be necessary to return information about the leg path – so that information about access paths and connection paths is linked to the physical restrictions that apply to them (ramps, escalators, lifts, stairs, tactile guidance strips, etc).

5.2.4 Seeking route information that can be displayed on maps

A trip plan will comprise several distinct components. At a minimum it will normally involve an access leg to get to the public transport system, a public transport leg and a final access leg to get from the public transport system to the required destination of the passenger's trip. Whilst the passenger should be able to rely on the public transport vehicle taking them from the boarding point to the alighting point, many passengers for one reason or another like to follow their route on a map. But the requirement for a map is much greater for the access legs at each end of a trip – and for any transfers necessary during the trip. To that end it should be possible to request sufficient details of the track (sequenced map coordinates) of the trip to be made such that it can be plotted over a topographic map of the trip at various scales or zoom levels, sufficient to see detail where it is most needed but also to be able to see the whole trip in context.

5.2.5 Refine a trip

An existing trip plan can be enriched with additional information, by one or several of the distributed planners:

- real-time information
- disruptions
- fares (using indirectly the fare service)
- additional information on vehicles
- additional information about facilities
- more detailed route information (e.g., tracks, path links)

The result is a refined trip plan or several refined trip plans.

Sometimes the enrichment may break the trip, then a trip recalculation is necessary. It is therefore important for the client to check, whether the result of a refinement is still a valid trip. The recalculation needs to be explicitly requested by the enquirer.

Voluntarily changing a trip is not considered a refinement in this context. For that purpose, the trip service or the trip changing service can be used to calculate or recalculate a part of a trip:

- Recalculation by changing the (sub-)mode of some legs
- Shortening a trip leg (e.g., leave one stop earlier)
- Changing the arrival or departure point of a trip leg
- Availability checks are to be done with the availability service.
- Searching with different conditions (e.g., with new accessibility requirements)
- Prolonging an interchange
- If a trip plan is based on estimated times only (in the case of shared mobility services, for instance), it has to be finalised using real time data and availability information.

5.2.6 Changing a Trip

While the trip refinement does not change the underlying trip, but only adds information to it, enquirers may want to use the trip as a template that they can adapt according to their personal preferences. To achieve that, an existing trip plan can be changed after its calculation in the following ways:

- Prolonging an interchange at a stop while retaining the trip before the interchange. If an enquirer prefers having more time at disposal at a particular interchange, she can ask for additional waiting time at the stop and specify that resulting changes shall be propagated towards the later part of the trip. The DJP will use that information to retain the trip until the modified interchange and search from there on for new solutions until the final destinations.
- Prolonging an interchange at a stop while retaining the trip after the interchange. The enquirer can request an additional waiting time for a particular stop while keeping the later part of the trip unchanged. The changes are propagated towards the beginning of the trip, resulting in an earlier start of the trip.
- This may also lead to a recalculation of the trip before or after the estimated trip leg (depending on whether the enquirer specifies that the changes shall be propagated towards the front or the back of the trip).

5.3 Additional tasks for a Distributed Journey Planning system

5.3.1 General

The API supports the following additional tasks. More detailed descriptions of all capabilities can be found in chapter 7.

5.3.2 Requesting a stop timetable

Whilst a traveller may wish to plan a specific trip, it may equally be useful to be able to see a list of all departures from a specific stop within a specified time window – a "stop timetable". A request for such a timetable might specify not only the stop but possibly an individual platform within such a stop.

5.3.3 Requesting times and other information for all intermediate stops in a trip

While a trip may be specified by simply its boarding and alighting stops and times, it is also possible to ask for trips to be described with details of all stops between the boarding and alighting points and the times at which these stops are expected to be reached. This would help a passenger follow the progress of their trip along the route.

Other information may be quay, occupancy, and capacity of a vehicle.

5.3.4 Requesting expected events at a particular stop

With an increasing number of operations having real-time monitoring and prediction systems the departure times of vehicle journeys can be obtained not only in the form of a timetable, but also in the form of predicted departure times calculated in real time. So, whilst this information is not available when pre-planning a trip a long time in advance, it becomes available close to the scheduled departure times (and becomes increasingly accurate as the scheduled time gets ever closer). A "stop event" request and response should deliver the best information known at the current time for departures from a stop in the next short while – perhaps 30 min or an hour. Whilst this information may be available locally, the DJP system provides an opportunity for such information to be available in the user's native language and in a style of presentation with which they are familiar if they make the request through the initiating system rather than seeking the information from a local information system.

Also information like occupancy may be shown.

5.3.5 Requesting information about a given journey or vehicle

More detailed information about a leg may be obtained through the journey or vehicle of that leg. This may include delays, quays, changes, formation, occupancy, vehicle features and capacity.

5.3.6 Requesting information about the fares and ticket options for a particular trip

One of the most challenging aspects of public transport information is that related to fares and tickets, with significant differences in the way in which fares are determined, in the nature of the types of ticket that are available, and the way in which tickets can be purchased. As journey planning systems increasingly provide at least basic information about fares and tickets so a DJP system should be able to interrogate such systems and convey the relevant information to enquirers. The priority for detailed information would be to give the price for a specific planned trip on the assumption that it is paid for with single trip tickets. However, such information places public transport at a disadvantage as many public transport systems have much cheaper ticket options for return tickets, multi-trip tickets, multi-person tickets, etc. So, a second priority would be to provide more general information about ticketing systems in a specific geographical area. In addition, there may be a requirement to provide hyperlinks to on-line retailers of relevant tickets.

5.3.7 Requesting information about lines

For a given line the enquirer may want to know the possible routes to display them to the passengers.

5.3.8 Requesting information about availability of journeys/vehicles

TRIAS allows for legs to be reserved and booked. As a first prerequisite the availability of a leg or legs needs to be ensured. E.g., can a taxi be at the address at 12:00 tomorrow? When do you need to be at the stop of the demand responsive bus? In some cases, the origin or destination of the leg may shift. E.g., when the pick-up point of the demand responsive bus is different from what was assumed by the trip planner in the first steps of the calculation. A recheck of the integrity of the trip is needed.

OJP is currently not intended for reservation or booking, and the availability service may be used many times more during the search for optimal trips. Therefore, implementers of the service should not use

internally the booking process to provide availability. The reason is that the booking process usually is very complex and involves locking possible journeys for a given time (especially for demand responsive buses). Therefore, the booking system may be easily overwhelmed by the trip planner, e.g., already by having to provide several variants for the same trip. This also means that the result of the availability service is usually volatile.

The availability service endpoint will often not be at the trip planner, but at the operator's site.

Availability may be most useful for:

- Taxi
- Shared modes of transport
- Demand responsive transport offers
- Other forms of vehicle pooling
- Availability of equipment or services for special needs

5.3.9 Other possible questions

Whilst the Open API will cover all normal forms of journey planning requests, it is recognised that some journey planning systems may wish to ask for supplementary information if it is available. Examples of such requests might be to seek a "rain safe" trip (one which minimises exposure to the weather), or to ask for an estimate of energy usage for each leg of a trip. Such requests could be added to the schema on a bi-partite basis between systems that can exchange such additional information, without undermining the generality of the Open API specification.

5.4 Advanced concepts/considerations

5.4.1 General

Some advanced considerations should be considered. Some parts will be implemented in the standard and some need to be taken care of by the home systems. Even then, a similar handling is advisable.

5.4.2 Routing for passengers with special needs

For passengers with special needs due to mobility or sensory impairments, it has to be possible to adapt the routes accordingly – e.g., by avoiding stairs and steps, preferring short walking distances, or trying to find transfers with tactile guidance strips. Filter options like *NoSight*, *NoStairs*, or *LevelEntrance* allow for filtering out any routes not satisfying such (hard) constraints. A problem with this approach, however, is that by imposing such hard constraints, one might either not find any route at all, or the route might be much less optimal in some other aspect (e.g., much longer travel time in order to avoid a single step). Another problem is that if one wanted to loosen a certain constraint (e.g., avoiding single steps) one would have to completely abandon it (e.g., it is not possible to express that one would prefer 1 step over 10 steps).

These limitations can partially be overcome by using the optimisation methods *leastWalking*, *minStairs*, *minLevelEntrances*, *bestForVisualImpairment*, *bestForAuditoryImpairment*. These optimisation methods are not guaranteed to be supported by a responding system and it is in the responsibility of the home system or the client application to translate the particular special needs into a series of TripRequest enquiries. A few examples may illustrate the approach.

Unaccompanied wheelchair. A trip is guaranteed to be feasible by wheelchair if the following filter options are chosen: *NoStairs*, *NoSingleStep*, *NoEscalator*, *NoTravelator*, *LevelEntranceOrBoardingAid*. An

enquiry using the optimisation method *earliestArrival* and one using *leastWalking* will result in a set of trips to choose among.

Walking frame. A person with a walking frame is typically slow and prefers short walking distances and a small number of interchanges. She may or may not be able to use the stairs. Let's assume the case that stairs are feasible but that they represent major obstacles that should be minimised. To find a relevant set of trips, the following sequence of enquiries might be used. Note that other sequences could be used as well, yielding the same or different results.

1. Optimisation method *earliestArrival*, reduced walking speed (to find the fastest variant that does not consider any of the soft constraints)
2. Optimisation method *leastWalking*, filter options *NoStairs*, *NoSingleStep*, *NoEscalator*, *LevelEntrance* (and keeping the reduced walking speed); if this does not return any result or if it is considerably longer than the fastest variant found previously, then further enquiries may follow:
3. Optimisation method *leastWalking*, filter options *NoStairs*, *NoEscalator* (allowing for single steps)
4. Optimisation method *leastWalking*, filter option *NoStairs*
5. Optimisation method *minStairs*
6. Optimisation method *minStairs*, filter option *NoEscalator*
7. Optimisation method *minStairs*, filter options *NoEscalator*, *LevelEntrance*
8. Optimisation method *leastChanges*
9. Optimisation method *leastChanges*, filter option *LevelEntrance*

Visual impairment. The filter option *NoSight* requires tactile guidance strips and assumes a slower walking speed. Thus, the following enquiries may be used:

- a) Optimisation method *earliestArrival*, filter option *NoSight*
- b) Optimisation method *leastWalking*, filter option *NoSight*

The optimisation method *bestForVisualImpairment* allows for additionally preferring platforms and vehicles with auditory signals:

- c) Optimisation method *bestForVisualImpairment*, filter option *NoSight*

If tactile guidance strips are not a hard constraint for the passenger, the following enquiry will give a result if the previous enquiries didn't find any route; or it may be able to find a faster route:

- d) Optimisation method *bestForVisualImpairment*

Handling of multiple impairments is also possible to some extent – consider, e.g., optimisation method *minStairs*, filter option *NoSight*, or optimisation method *bestForAuditoryImpairment*, filter options *NoStairs*, *NoSingleStep*, *LevelEntrance*.

5.4.3 Via, fixed legs

Vias and fixed legs are supported. However, some trip planners are not able to handle one or multiple vias. The solution in that the enquirer handles such cases: A via can always be seen as two trips with

(some) waiting time in between. Fixed legs (e.g.- if a plane was booked weeks ago) can also be seen as something where three trips need to be handled and concatenated in the end. The same may be a solution for some other subjects that will be treated further below (e.g., see 5.4.5 or 5.4.8). As with some cases of trip refinement the whole concatenated trip needs to be validated and sometimes recalculated (e.g., - when the plane arrives an hour late). OJP in such cases puts the burden to keep those things together to the enquirer.

5.4.4 User preferences

5.4.4.1 General

Mobility has undergone deep, disruptive changes over the last years. Major trends such as urbanisation, digitalisation, "mobile first", sharing economy and decarbonisation have triggered hype trends such as MaaS (Mobility as a Service) or ABT (Account Based Ticketing) and new mobility modes such as micro-mobility, ride-hailing, carpooling or on-demand transport.

Trip planners in web apps and smartphone apps have become a crucial element of this transformation. They have evolved from restricted, single-mode route planners or public transport planners into more and more comprehensive, detail-rich, intermodal, end-to-end trip planners.

However, intermodality and comprehensiveness add a new challenge: trips may now be composed of a vast choice of modes and operators, not only on the first and last mile, but also on the main legs of the trip. A truly open, unbiased, intermodal trip planner algorithm could compute thousands or millions of combinations for a trip from A to B, instead of just a few combinations that are typical for a simple, old-fashioned PT planner.

Thus, trip planners are faced with the problem of an overwhelming number of choices:

- Which of the many combinations should be presented to the user?
- On what grounds should some results be displayed and many other results not?

5.4.4.2 Definition

User preferences are a range of criteria or requirements which express what a user prefers when faced with a large choice of options. Each preference gives an indication about how suitable a certain option is for the given user. Each user preference can be expressed as a criterion (e.g., "sportiness") or as a question (e.g., "how important is environmental-friendly travel for you?"). The actual preference may be expressed on a scale, typically in a few well-defined levels (e.g., "very important", "average", "not so important", or "yes"/"no"). User preferences may be influenced somewhat by marketing and trends, but at the core, they are very much an individual, private matter. Only the user himself or herself can actually determine what his or her preferences actually are.

USER PREFERENCE could be added to Transmodel as an additional concept to take this into account

5.4.4.3 Usage

Various preferences may be combined through Boolean AND-operations, e.g., "sportiness is average AND environmentally friendly is very important". A given statement of user preferences allows to narrow down the choice of suitable options. With a good set of user preferences, the choice may be boiled down to just a few options. Furthermore, the options can be scored and compared based on the preferences, and only the options with the highest scores need to be displayed to the user.

5.4.4.4 Good user preferences

Good preferences must be relevant, easy to understand, clear and unambiguous. The actual preference should be expressed either as a binary yes/no, or quantified on a short scale in 3, 4 or 5 steps with well-defined meanings, so that they can be easily understood and applied by most users. A greater number of steps is both harder to respond to by users and more difficult to implement in an IT system.

5.4.4.5 Mechanism

High level User Preferences will have to be transformed to low level search parameters on the different engines. In OJP many of the low-level parameters already exist. In some cases, the User Preference is vague, because it is defined by emotions and beliefs and is hard to express in a concrete query. The availability of data poses another problem.

We present here some ideas how a possible User Preference scheme can be implemented given the low-level parameters. This may be used by developers of responding systems as a reference about how to handle them.

5.4.4.6 Suggested user preferences

The following list of User Preferences was studied and used to identify new low-level parameters.

Table 1 — User Preferences and their implementation

User Preference	Description and Possible values	Low Level Parameter, Handling and Qualification
LowPrice	User's preference that the trip has a low price. Values: important, average, unimportant	- Origin & Destination & monomodal IndividualTransportOptions = all modes, except taxi - OptimisationMethod=leastCost - Verdict: The currently defined low-level parameters can cover this user preference.
Speed	User's preference about the speed/short duration of the trip Values: important, average, unimportant	- Origin & Destination & monomodal IndividualTransportOptions = all modes - OptimisationMethod=fastest - Verdict: The currently defined low-level parameters can cover this user preference.
Comfort	User's preference about the comfort/convenience of the trip, including few changes, short transfers, and convenient vehicle types. Reflect USER PREFERENCE from Transmodel Values:	- Two TripRequests with: - OptimisationMethod=minC hanges - OptimisationMethod=least Walking - Origin & Destination & monomodal IndividualTransportOption= taxi - The convenient vehicle type part is not clear. This part is not managed in a good way with the current low-level parameters

User Preference	Description and Possible values	Low Level Parameter, Handling and Qualification
Sportiness	User's endurance and fitness for sportive, self-powered trip legs (walk, cycle, etc.) Values: high, average, low	- Origin & Destination & monomodal IndividualTransportOption=bicycle - increase MaxDistance for Cycle - WalkSpeed=150% - Verdict: The currently defined low-level parameters can cover this user preference.
EnvironmentalSafety	User's preference about climate/environmental friendliness of the trip (option 'low' might have no effect) Values: high, average, low	- OptimisationMethod=environmentalSafety (Minimized carbon footprint) - Some aspects of environmental-friendly might not be covered by the CO2 footprint alone.
WeatherProtection	User's preference not to be exposed to eventual weather hazards, thus, to avoid walks and micro mobility. Values: true/false	- OptimisationMethod=leastWalking - Origin & Destination & monomodal IndividualTransportOption=car - Some aspects of the User Preference may not be covered by the current low-level parameters (e.g., does it depend on a covered waiting place?)
ExtraSafe	User's preference for an extra high level of safety Values: true/false	- OptimisationMethod=minChanges - Origin & Destination & monomodal IndividualTransportOption=car - OptimisationMethod=extraSafe - The definition allows currently for a big interpretation space.
ExtraReliable	User's preference for an extra high reliability (low probability of delays, cancellations, etc.) Values: true/false	- Origin & Destination & monomodal IndividualTransportOption=car, bicycle (no AlternativeModesOfOperation since they may no longer be available later.) - TransferType=guaranteedConnection, protectedConnection - WalkSpeed=60% - PtModeFilter= only train - OptimisationMethod=extraReliable

User Preference	Description and Possible values	Low Level Parameter, Handling and Qualification
		- No Journey with bad prognosis quality - AdditionalTransferTime=absolute Time - Verdict: The currently defined low-level parameters can cover this user preference.
ExtraAccessible	User's preference for perfect accessibility (lifts, ramps, no stairs, etc.), including extra transfer time, short transfer distances, etc.. Values: true/false	- NoSingleStep=true - NoStairs=true - NoEscalator=true - NoElevator=false - NoRamp=false - LevelEntrance=true - WalkSpeed=60% - A rules engine may be needed if no trip could be found. Otherwise, low-level parameters handle the User Preference.
Scenic	User's preference for scenic, touristic impressions on some parts of the trip. Values: true/false	- OptimisationMethod=Scenic - Low level parameters manage this User Preference.
QuietTravel	User's preference for quiet transport (quiet travel compartments or low occupation) Values: true/false	- OptimisationMethod=Peaceful (only formation information with 1. Class or quiet part currently possible) - Verdict: The currently defined low-level parameters can cover this user preference.
WalkingProfile	Users walking profile (especially for hiking) Values: normal, medium, difficult	- HikingProfile=normal, intermediate, difficult] - Verdict: The currently defined low-level parameters can cover this user preference.

User Preference	Description and Possible values	Low Level Parameter, Handling and Qualification
CyclingProfile	Users cycling profile (especially for sportive activities) Values: normal, fast, green, comfortable	- CyclingProfile=normal, fast, green - Verdict: The currently defined low-level parameters can cover this user preference.

5.4.5 IncludeAlternativeOptions in trips

In many cases it is unclear which information should be really provided as a result. An example is two possible bike routes or an additional scenic trip. What exactly is provided by the service can't necessarily be described beforehand and may depend on intentions of the responding system (e.g., it wants to sell special tickets, it believes that some parameters were not set correctly). E.g., for classic PT-Trips these alternative options could be longer trips with less interchanges, although the optimisation method shortest trip was requested. In a special parameter such additions can be encouraged, the results should be marked as containing such additions.

5.4.6 Handling of slow traffic

In many cases slow traffic plays a particular role and especially transport operators want to show them even when they don't fulfil the optimisation criteria (e.g., fastest and least transfers).

Several possibilities can be used to show them with parameters provided by OJP:

- Exclude the dominant operator in a secondary search for one or more trip legs. Doing the search for the whole trip with filters may cause more "harm" when also a crucial part by one operator is then done by a different operator. An example is the SOB train from Bern to Chur, which is slower, but one never needs to change. Excluding SBB will work fine to find the SOB train. However, if one searches from Lausanne to Chur and excludes SBB, then the trip may become very strange (see Figure 2 and Figure 3).
- Exclude a dominant mode in a secondary search for one or more trip legs.
- Setting an artificial via point to force e.g., a mountain track (when a tunnel is available).
- Analyse the trip result and if e.g., two stops are at the same lake, then do a secondary search with a boat trip in between.
- Use a trip refinement to a special server that gives you possible updates. This is not the standard behaviour of a trip refinement service and needs to be implemented differently.

Some trip planners might have proprietary parameters to add slow traffic. However, this is not covered in this specification.



Figure 2 – Searching with SBB (left, one transfer in Zürich, fast) and without (right, SOB direct train)

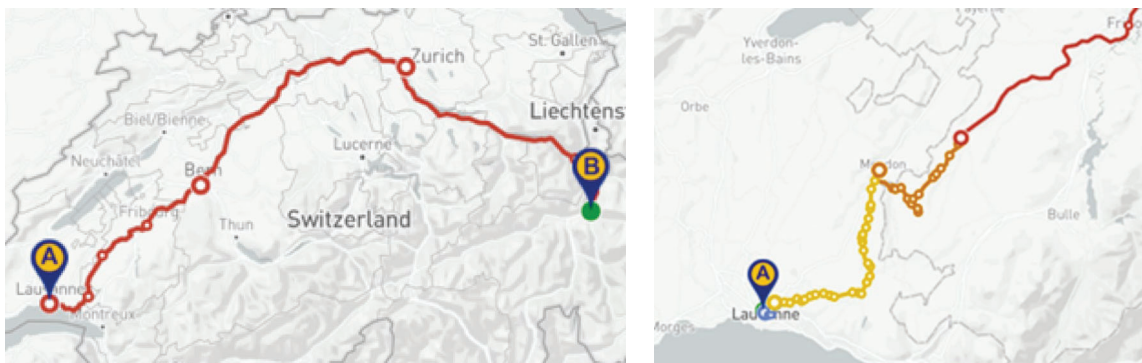


Figure 3 – Searching with SBB from Lausanne to Chur (left, two transfers, fast) and without (right, three different buses without even reaching Bern yet). The right image on purpose only shows part of the trip.

5.4.7 Demand responsive transport

5.4.7.1 General

Some demand responsive transport can be modelled as simple journeys. However, if area oriented public transport is in play, some additional aspects must be covered by the use cases and in the way they are fulfilled by the services. A smooth and reliable TRIP presentation is important for the success of on-demand offers in public transport.

This results in the following necessary adaptations in how the OJP interface is used:

- Assumptions regarding the behaviour of the Leg must be modelled (based on the rules that govern such a FLEXIBLE LINE or MOBILITY SERVICE).
- The location service should show which OPERATOR or MODE are available and also allow presentations on a map in some sort (e.g., showing an icon indicating on-demand offers at the location pointer, showing active vehicles of an operator, polygons of operator area, icons on stops).
- The line information service should provide information about the area(s), VEHICLE MEETING POINTS and rules that govern such a service (e.g., for the above mentioned presentation on maps),