

STN P	Inteligentné dopravné systémy Verejná doprava Otvorené aplikačné programovacie rozhrania (API) pre distribuované plánovanie cestovania	STN P CEN/TS 17118 01 8613
------------------	---	--

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

Táto norma obsahuje anglickú verziu európskej normy.
This standard includes the English version of the European Standard.

Táto norma bola oznámená vo Vestníku ÚNMS SR č. 03/25

Táto predbežná slovenská technická norma je určená na overenie. Prípadné pripomienky pošlite do decembra 2026 Úradu pre normalizáciu, metrológiu a skúšobníctvo Slovenskej republiky.

Obsahuje: CEN/TS 17118:2024

Oznámením tejto normy sa ruší
STN P CEN/TS 17118 (01 8613) z marca 2019

140146



TECHNICAL SPECIFICATION

CEN/TS 17118

SPÉCIFICATION TECHNIQUE

TECHNISCHE SPEZIFIKATION

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**

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

CEN/TS 17118:2024 (E)

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	<i>OJP 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	<i>OJPRequestStructure</i>	94
8.3.2.3	<i>OJPResponseStructure</i>	95
8.3.2.4	<i>ServiceRequestStructure</i> and <i>ServiceRequestContext</i> (from SIRI).....	96
8.3.2.5	<i>ServiceDeliveryStructure</i> (from SIRI).....	100
8.3.3	<i>OJP_UTILITY</i>	101
8.3.3.1	General	101
8.3.3.2	Simple types.....	101
8.3.3.3	<i>InternationalTextStructure</i>	101
8.3.3.4	<i>WebLinkStructure</i>	102
8.3.4	<i>OJP_ModesSupport</i>	102
8.3.4.1	General	102
8.3.4.2	Simple types.....	102
8.3.4.3	<i>IndividualTransportOptionStructure</i>	107
8.3.4.4	<i>ItModesStructure</i>	108
8.3.4.5	<i>ModeStructure</i>	108
8.3.4.6	<i>ModeAndModeOfOperationFilterStructure</i>	109
8.3.4.7	<i>ModeFilterStructure</i>	110
8.3.5	<i>OJP_Common</i>	111
8.3.5.1	Simple types.....	111
8.3.5.2	<i>OJPError</i>	111
8.3.5.3	<i>OJPErrorStructure</i>	112
8.3.5.4	<i>ErrorType</i>	112
8.3.5.5	<i>PrivateCodeStructure</i>	112
8.3.5.6	<i>LinearShapeStructure</i>	112
8.3.5.7	<i>AreaStructure</i>	112

CEN/TS 17118:2024 (E)

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

CEN/TS 17118:2024 (E)

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	LegAlignStructure	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

CEN/TS 17118:2024 (E)

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

CEN/TS 17118:2024 (E)**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

CEN/TS 17118:2024 (E)

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/>

CEN/TS 17118:2024 (E)**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.

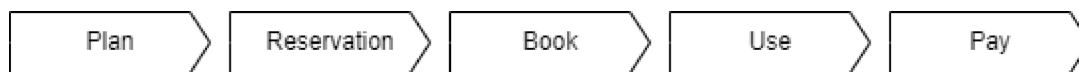


Figure 1 — Processes around travel

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.

koniec náhľadu – text ďalej pokračuje v platenej verzii STN