Automotive Requirements for SPaT and MAP

CAR 2 CAR Communication Consortium

About the C2C-CC

Enhancing road safety and traffic efficiency by means of Cooperative Intelligent Transport Systems and Services (C-ITS) is the dedicated goal of the CAR 2 CAR Communication Consortium. The industrial driven, non-commercial association was founded in 2002 by vehicle manufacturers affiliated with the idea of cooperative road traffic based on Vehicle-to-Vehicle Communications (V2V) and supported by Vehicle-to-Infrastructure Communications (V2I). The Consortium members represent worldwide major vehicle manufactures, equipment suppliers and research organisations.

Over the years, the CAR 2 CAR Communication Consortium has evolved to be one of the key players in preparing the initial deployment of C-ITS in Europe and the subsequent innovation phases. CAR 2 CAR members focus on wireless V2V communication applications based on ITS-G5 and concentrate all efforts on creating standards to ensure the interoperability of cooperative systems, spanning all vehicle classes across borders and brands. As a key contributor, the CAR 2 CAR Communication Consortium works in close cooperation with the European and international standardisation organisations such as ETSI and CEN.

Disclaimer

Document information

Number:	2077	**Version:**	n.a.	**Date:**	12/03/2021
Title:	Automotive Requirements for SPaT and MAP			**Document Type:**	RS
Release	1.5.3				
Release Status:	Public				
Status:	Obsolete				

Table 1: Document information

Changes since last version

Title:	Automotive Requirements for SPaT and MAP		
Date	**Changes**	**Edited by**	**Approved**
12/03/2021	Mo changes (still obsolete)	Release Management	Steering Committee
16/12/2020	Set to obsolete	Release Management	Steering Committee
31/07/2020	No changes	Release Management	Steering Committee
27/03/2020	Initial release	Release Management	Steering Committee

Table 2: Changes since last version

Table of contents

List of tables

1 Introduction

Other (informational) **RS_ARSM_1**

This document was a working document in 2019, containing identified requirements to the SPATEM and MAPEM from an automotive perspective.

It served as an extension to already existing requirements on SPATEM and MAPEM in the C-Roads profiles.

The contained requirements have been incorporated in C-Roads Release 1.7 from July 2020 for the most part. Therefore, this document will not be maintained further and the C-Roads "C-ITS Message Profiles and Parameters" document shall be used as single reference for the most up-to-date requirements on MAPEM and SPATEM.

2 Scope

Other (informational) **RS_ARSM_2**

The present document provides requirements related to the features of a C-ITS station transmitting SPATEM and MAPEM to enable interoperability.

In some cases, requirements are written in a way which let the implementation open, for example if they refer to very specific implementations which may depend on specific national regulations. Those requirements have to be further detailed by anybody implementing that requirement. Beside these special requirements all other requirements can be further detailed, too.

3 Conventions used

Other (informational) (RS_BSP_152) **RS_ ARSM_3**

Conventions used in this and other C2C-CC specification documents can be found in [C2CCC ConV].

4 Definitions

Definition (RS_BSP_193) **RS_ARSM_81**

'C-ITS time' or 'time base' means the number of elapsed International Atomic Time (TAI) milliseconds since 2004-01-01 00:00:00.000 Coordinated Universal Time (UTC)+0 as defined in [EN 302 636-4-1]. Timestamps as defined in [TS 102 894-2] follow this time format.

Definition **RS_ARSM_82**

The '*station clock'* means a clock representing Cooperative Intelligent Transport Systems (C-ITS) time in a C-ITS station (see RS_RSP_006).

Definition (RS_BSP_429) **RS_ARSM_100**

Information provided with a *'confidence level'* of 95 % means that the true value is inside the confidence interval or the confidence area for at least 95 % of the data points in a given statistical population.

Definition (RS_BSP_500) **RS_ARSM_87**

A *'confidence interval'* is specified by the estimated value plus/minus the confidence value.

Definition **RS_ARSM_93**

An *'instant'* denotes a point on the time axis, often also referred as a 'moment in time' (see also IEC 60050).

Definition **RS_ARSM_6**

A *'merge point'* designates a node of a lane where the lane is split into two lanes in driving direction towards an intersection (ingress). This is due to the fact, that according to C-Roads definitions and the Delegated Act, Annex II, chapter 3.7.3 'Road Lane Topology (RLT) service', all attributes are provided in the order of the nodes.

On the other hand, a merge point on an egress lane is located, where two lanes end in one lane in driving direction.

The opposite applies for *'diverge points'*.

Definition **RS_ARSM_7**

A *'conflict area'* is the area of the intersection that is limited by the first nodes of ingress / egress vehicle lanes, first nodes of 'ingresspath' crosswalk lanes, and stop lines of bicycle lanes. For a better understanding, see also e.g. Figure G.6 in [ISO/TS 19091].

Definition **RS_ARSM_95**

'Phase' is a general term denoting all the movement phase states strictly allowing or prohibiting to proceed into an intersection (so the 'Reds' and 'Greens' as summarized in SAE J2735).

Definition **RS_ARSM_96**

'Transition' is a general term denoting all the movement phase states which are not covered by the term phase (so the 'Yellows / Ambers' as summarized in SAE J2735).

Definition **RS_ARSM_97**

'Phase state' is a general term that covers all movement phase states as defined in SAE J2735, i.e. 'phase state' includes both phases and transitions.

Definition **RS_ARSM_116**

'Actuated traffic light operation' refers to an operation mode of the traffic light controller that dynamically adapts the changes to the current traffic situation (i.e., the cycle of the traffic phases is not static but may change over time).

Definition **RS_ARSM_109**

A traffic light is considered *'operational'*, if the corresponding traffic light controller is neither switched off nor in any kind of failure mode. This means that also traffic lights showing some kind of 'standby' (e.g. at night) are considered operational.

5 Parameter settings

Definition **RS_ARSM_443**

Table 3: Parameter settings

Parameter	Value	Unit	Description	Min. Value	Max. Value	Source Document
tMapCompleteTransmission	1	s	Time duration within the whole MAP including all fragments shall be transmitted	--	--	--
dRangeIdUnique	5	km	Radius around every intersection within which the IntersectionID tuple shall be unique	5	--	--
pLateralNodeOffset	3	m	Maximum lateral offset to the center of the lane for the node points within a MAP	--	--	--
*pLateralNodeOffset*AD	1	m	Maximum lateral offset to the center of the lane for the node points within a MAP if automated driving shall be supported	--	--	--
pLaneAngleDeviation	5	°	Maximum angle between the connection of the node points and the corresponding tangent to the lane center	--	--	--
pMaxPerpendDistLaneCenter	3	m	Maximum perpendicular distance between the linear connection of two consecutive lane nodes and the actual center of the lane	--	--	--
pMaxNoOfNodesPerLane	18	--	Maximum allowed number of nodes per lane	--	--	--
pMinLaneWidth	2,6	m	Minimum width of a merging/diverging lane before enabling/disabling the taper to left / right indication	--	--	--
pMinIngressLaneLength	300	m	Minimum length of an ingress lane representation in MAPEM	--	--	--

Parameter	Value	Unit	Description	Min. Value	Max. Value	Source Document
pSpeedLimitHigh	60	kph	Allowed speed limit above which the required minimum ingress lane length is increased	--	--	--
pMinIngressLaneLengthHighSpeed	500	m	Minimum length of an ingress lane representation in MAPEM for an allowed speed limit above *pSpeedLimitHigh*	--	--	--
pMinEgressLaneLength	5	m	Minimum length of an egress lane representation in MAPEM	--	--	--
tSubSystemClockAccuracy	200	ms	Accuracy of the system clock of the subsystem responsible for the generation of time change details	--	--	--
tIntraSystemClockAccuracy	500	ms	Maximum deviation between the different system clocks	--	--	--
pSpatUpdateDelay	100	ms	Maximum period between SPaTEM update in content and its transmission	--	--	--
fSpatTransmissionFreq	10	Hz	Transmission frequency for SPaT messages	--	--	--
pTimeMarkUnknown	36 001	1/10 s	Value to indicate an unknown TimeMark	--	--	[SAE J2735]
pTimeMarkMin	0	1/10 s	Minimum value of TimeMark	--	--	[SAE J2735]
pTimeMarkOutOfRange	36 000	1/10 s	Value to indicate an instant which is not in the UTC hour of the referenced instant	--	--	[SAE J2735]
tTimeOfChangeAccuracy	500	ms	Accuracy of phase state change time information for signal controllers operating at fixed time	--	--	--
tTimeChangeInterval	1,5	s	Interval to be used for the calculation of the likely time confidence	--	--	--

Parameter	Value	Unit	Description	Min. Value	Max. Value	Source Document
tDelayFailureTransmission	200	ms	Maximum allowed delay between the instant the traffic light controller goes into failure until the information is being transmitted	--	--	--

6 Requirement specifications

6.1 MAPEM Automotive Requirements

6.1.1 Transmission

Other (informational) **RS_ARSM_84**

The following requirement on MAPEM apply in addition to the relevant standards ([TS 103 301], [ISO/TS 19091], [SAE J2735]).

Details:

Tested by:

Requirement **RS_ARSM_10**

If more than one MAP message is sent out for the according intersection (fragments) the complete MAP (including all fragments) shall be sent within *tMapCompleteTransmission*. In case this is not feasible while complying with DCC regulations, DCC regulations shall always prevail over this requirement.

Details:

Tested by:

6.1.2 IntersectionGeometry

Requirement **RS_ARSM_11**

The data field ‘id’ (DF_IntersectionReferenceID) shall consist of both ‘region’ (DE_RoadRegulatoryID) and ‘id’ (DE_IndersectionID)

Details:

Tested by:

Requirement **RS_ARSM_12**

The id tuple referred to in RS_ARSM_11 shall be unique within a radius of *dRangeIdUnique* around each intersection.

Details:

Tested by:

Requirement **RS_ARSM_13**

The id tuple referred to in RS_ARSM_11 shall be identical to the appropriate id tuple of the corresponding SPATEM 'intersectionState'.

Details:

Tested by:

Requirement **RS_ARSM_14**

The data element 'laneWidth' (DE_LaneWidth) shall be set mandatorily for every intersection.

Details:

Tested by:

Requirement **RS_ARSM_15**

The 'laneSet' (DF_LaneList) contained in the IntersectionGeometry shall include all vehicle lanes of an intersection and all other lanes of an intersection that have signalized connections (e.g. including lanes for pedestrians (crosswalk), cyclists (bikeLane), tracked vehicles (trackedVehicles) and busses (vehicle)).

Details:

Tested by:

6.1.3 GenericLane

Requirement **RS_ARSM_16**

For unidirectional lanes, exactly one of the data elements 'ingressApproach' and 'egressApproach' (of type DE_ApproachID) shall be present and used.

For further details on how to use these data elements, see ISO /TS 19091:2019 G.8.2.6.

Details:

Tested by:

Requirement **RS_ARSM_17**

For bidirectional lanes that cross both the ingress- and egress approach of an intersection arm (e.g. bike lanes or crosswalks) both data elements 'ingressApproach' and 'egressApproach (of type DE_ApproachID) shall be present and used to indicate the approaches that are crossed.

For further details on how to use these data elements, see ISO /TS 19091:2019 G.8.2.6.

Details:

Tested by:

Requirement **RS_ARSM_18**

For lanes of type vehicle (LaneAttributes.LaneTypeAttributes = vehicle), ingress lanes of a common drive direction towards the intersection shall have a common ingress approach ID as a mandatory attribute.

Details:

Tested by:

Requirement **RS_ARSM_117**

The data element ‘maneuvers’ (of type DE_AllowedManeuvers) shall not be present in any instance of a ‘generic lane’ within a MAPEM.

Note: The information about allowed maneuvers is contained in the instances of ‘ConnectingLane’ (see RS_ARSM_21). This requirement intends to avoid duplicate or possible inconsistencies.

Details:

Tested by:

6.1.4 NodeListXY

Requirement **RS_ARSM_118**

For all lanes represented in MAPEM within the data element nodeListXY (of type DF_NodeListXY) the data element nodes (a list of DF_NodeSetXY) shall be used.

This implies that the data element computed (of type DF_ComputedLane) shall not be used.

Details:

Tested by:

Requirement **RS_ARSM_25**

The first node of any vehicle lane shall be the node of the lane which is closest to the center of the intersection.

Details:

Tested by:

Requirement **RS_ARSM_26**

The first node of an ingress lane, which is not a diverge or merge point, shall be the node that shall not be passed by a vehicle when movement is not allowed (from regulations, typically this is the stop line on the street).

Note: This adds on to [ISO/TS 19091] where it is only stated that the first node 'should be the node closest to the geometric centre of the intersection, and is typically at the stop line'. This is only part of the informative text – see [ISO/TS 19091], 6.5.7.

Details:

Tested by:

Requirement **RS_ARSM_44**

The ingress lanes shall follow the main road with priority towards the intersection.

Note: If a non-priority road shall be included into the ingress structure, all lanes of the non-priority roads shall be grouped into one or several separate approaches that only represent these non-priority roads.

Details:

Tested by:

Requirement **RS_ARSM_40**

For each ingress approach at least one ingress lane of type vehicle shall be at least *pMinLaneIngressLength* long following the priority road.

Details:

Tested by:

Requirement **RS_ARSM_41**

If an adjacent intersection is closer than *pMinIngressLaneLength*, ingress lanes shall be shortened to the first egress point of the adjacent intersection. If no MAPEM is transmitted for the adjacent intersection, the ingress lanes shall be shortened such that they don't intersect the adjacent intersection's conflict area.

Details:

Tested by:

Requirement **RS_ARSM_43**

At intersections with higher speed limits allowed (> *pSpeedLimitHigh*) the ingress lanes shall be minimum *pMinIngressLaneLengthHighSpeed* long.

Details:

Tested by:

Requirement **RS_ARSM_42**

If the number of nodes of a lane exceeds *pMaxNoOfNodesPerLane* (see RS_ARSM_35) to fulfil RS_ARSM_34, the ingress lane might be shorter than *pMinIngressLaneLength* or *p_MinIngressLaneLengthHighSpeed*.

Details:

Tested by:

Requirement **RS_ARSM_47**

Vehicle egress lanes shall have a minimum length of *pMinEgressLaneLength*.

Details:

Tested by:

Other (informational) **RS_ARSM_31**

Node points should correspond to the center of the lane.

Details:

Tested by:

Requirement **RS_ARSM_32**

The absolute lateral offset of node points to the center of the lane shall be less than *pLateralNodeOffset*.

Details:

Tested by:

Requirement **RS_ARSM_94**

Let $\overline{a}$ be the vector representing the linear connection of two node points, and $\overline{p}$ be the vector representing the shortest distance of vector $\overline{a}$ to the center of the lane (that is, $\overline{p}$ is perpendicular to the tangent of the center line of the lane at the foot of the dropped perpendicular).

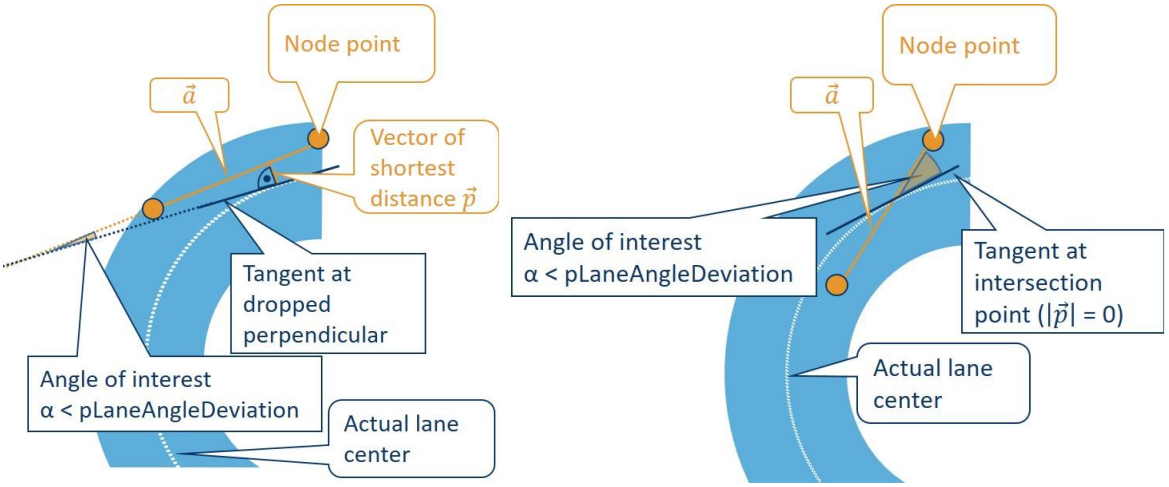
Then for $|\overline{p}| > 0$ it shall always hold that

$$\cos^{-1}\left(1 - \frac{\overline{a} * \overline{p}}{|\overline{a}| * |\overline{p}|}\right) \leq pLaneAngleDeviation.$$

For $|\overline{p}| = 0$ (i.e. $\overline{a}$ crosses the lane center) the angle α between $\overline{a}$ and the tangent to the lane center at the intersection point with the lane center shall be less than *pLaneAngleDeviation*.

Note: In less formal wording this means that the angle between the linear connection of two node points and the corresponding tangent to the lane center shall not be greater than *pLaneAngleDeviation.*

Note: See drawings below for a better understanding:



Details:

Tested by:

Requirement **RS_ARSM_34**

The perpendicular distance between the linear connection of two node points and the center of the lane shall be less than *pMaxPerpendDistLaneCenter*.

Details:

Tested by:

Requirement **RS_ARSM_35**

The number of node points shall be limited to *pMaxNoOfNodesPerLane* nodes per lane (for both ingress and egress lanes).

Details:

Tested by:

6.1.5 NodeAttributeSetXY

Requirement **RS_ARSM_36**

Each diverge or merge point (of type DF_NodeXY) shall be explicitly marked with corresponding node attribute (DF_NoteAttributeSetXY) ‘divergePoint’ or ‘mergePoint’ (for the definition see RS_ARSM_6).

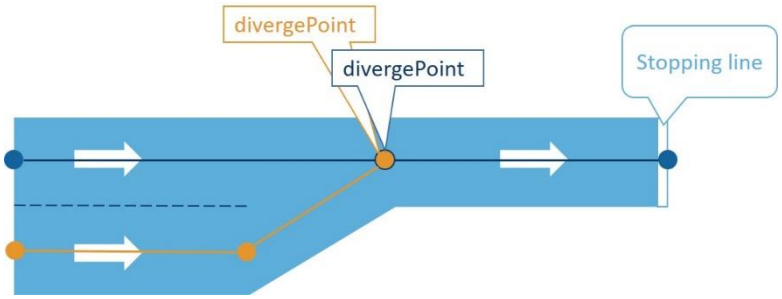
Note: For further details see [ISO/TS 19091] ‘localNode’.

Details:

Tested by:

Requirement **RS_ARSM_37**

For diverging / merging lanes one node shall be defined as diverge / merge point according to RS_ARSM_36. This node shall be present with the same position (relative to the intersection reference point) in the ongoing lane and as first / last node in the diverging / merging lane.



Details:

Tested by:

Requirement **RS_ARSM_27**

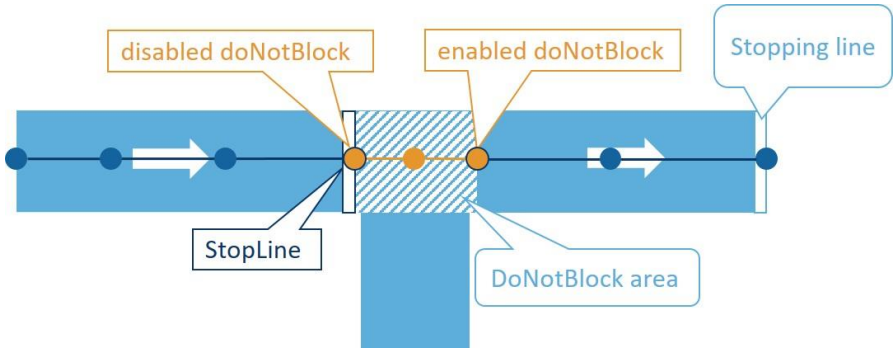
In the MAPEM an additional attribute ‘stop line’ exists. This shall be used for additional upstream ‘do not block’ lines (e.g. in Germany: ‘bei Rot hier halten’).

Details:

Tested by:

Other (informational) **RS_ARSM_28**

A SegmentAttributeXY of value ‘doNotBlock’ is intended to be present in the ‘enabled’ list for a road segment where a vehicle may not come to a stop. For detailed requirements see RS_ARSM_29 and RS_ARSM_30.



Details:

Tested by:

Requirement **RS_ARSM_29**

A SegmentAttributeXY of value ‘doNotBlock’ shall be present in the ‘enabled’ list at the first node of the lane that shall not be blocked by a vehicle in case of a queue in front of the traffic light.

Details:

Tested by:

Requirement **RS_ARSM_30**

A SegmentAttributeXY of value ‘doNotBlock’ shall be present in the ‘disabled’ list at the first node of the lane thereafter (see RS_ARSM_29), which may again be blocked by a vehicle.

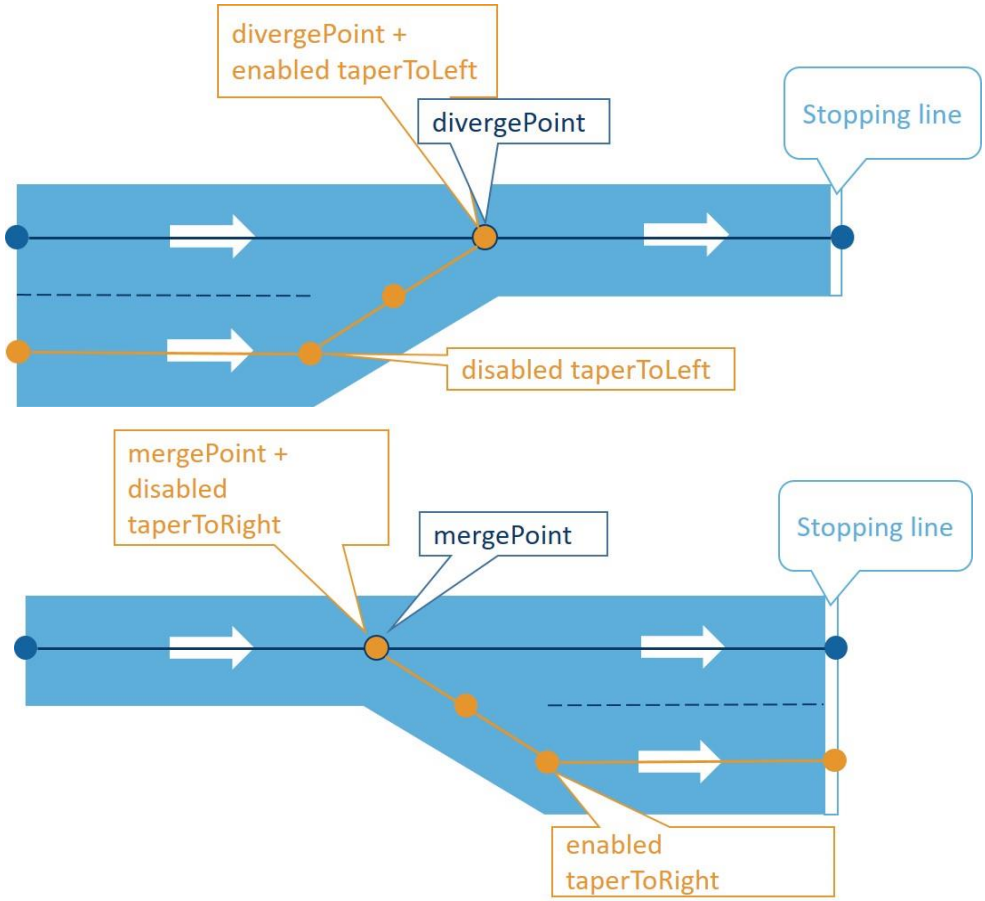
Details:

Tested by:

Other (informational) **RS_ARSM_111**

For diverging lanes, a SegmentAttributeXY of value ‘taperToLeft’ or ‘taperToRight’ is intended to be enabled at the diverge point (i.e. first node) if the lane width of the diverging lane is below *pMinLaneWidth* at the first node after the diverge point. It is intended to be disabled at the first node where the diverging lane has reached a width of at least *pMinLaneWidth*. For merging lanes, ‘tapeToLeft’ or ‘taperToRight’ is intended to be enabled at the last node where the merging lane has a width of at least pMinLaneWidth and disabled at the merge point.

For detailed requirements see RS_ARSM_38, RS_ARSM_112, RS_ARSM_113 and RS_ARSM_114.



Details:

Tested by:

Requirement **RS_ARSM_38**

A SegmentAttributeXY of value ‘taperToLeft’ or ‘taperToRight’ shall be present in the ‘enabled’ list of the first node of the diverging lane, if the lane width of the diverging lane is below *pMinLaneWidth* at the first node after the diverge point (i.e. more than two nodes are used to describe the tapering part of the diverging lane, e.g. for accuracy reasons).

Details:

Tested by:

Requirement **RS_ARSM_113**

A SegmentAttributeXY of value ‘taperToLeft’ or ‘taperToRight’ shall be present in the ‘enabled’ list of the last node of the merging lane, for which the merging lane has a lane width of at least *pMinLaneWidth*, if the lane width of the merging lane is below *pMinLaneWidth* at the last node

before the merge point (i.e. more than two nodes are used to describe the tapering part of the merging, e.g. for accuracy reasons).

Details:

Tested by:

Requirement **RS_ARSM_112**

If ‘taperToLeft’ or ‘taperToRight’ is present in the ‘enabled’ list of a previous node (as described in RS_ARSM_38), a SegmentAttributeXY of same value shall be present in the ´disabled´ list at the first node where the diverging lane has reached a width of at least *pMinLaneWidth*.

Details:

Tested by:

Requirement **RS_ARSM_114**

If ‘taperToLeft’ or ‘taperToRight’ is present in the ‘enabled’ list of a previous node (as described in RS_ARSM_113), a SegmentAttributeXY of same value shall be present in the ´disabled´ list at the last node of the merging lane.

Details:

Tested by:

Requirement **RS_ARSM_39**

If a diverging / merging lane has a width of at least *pMinLaneWidth* at the first / last node after / before the diverge point / merge point, a SegmentAttributeXY of value ‘taperToLeft’ or ‘taperToRight’ may be present in the ‘enabled’ and ‘disabled’ list of the respective nodes (as specified in RS_ARSM_38, RS_ARSM_112, RS_ARSM_113 and RS_ARSM_114).

If present, it shall be used as described in the aforementioned requirements.

Details:

Tested by:

6.1.6 ConnectsToList

Requirement **RS_ARSM_119**

The data element ‘connectsTo’ (of type DF_ConnectsToList) shall be present at least for every ingress lane of the intersection that is controlled by a traffic light.

Details:

Tested by:

Requirement **RS_ARSM_19**

The data field ‘connectsTo’ (DF_ConnectsToList) shall include every possible connection between ingress and egress lanes of one intersection – u-turns optional if they are allowed by traffic rules. The contained connections shall however not include those requiring lane changes in the conflict area (if applicable).

Details:

Tested by:

Requirement **RS_ARSM_20**

There shall be no duplicate connections indicated via ‘connectsTo’ between the same ingress and egress lanes for the same direction.

Details:

Tested by:

Requirement **RS_ARSM_48**

The data element ‘signalGroup’ (DE_SignalGroupID) shall be given for every connection that is signalized with at least one operational traffic light (see also RS_ARSM_74).

Details:

Tested by:

Requirement **RS_ARSM_49**

Every given ‘signalGroup’ / ‘intersectionReferenceID’ tuple in the MAPEM shall also be found in the SPATEM.

Details:

Tested by:

6.1.7 AllowedManeuvers in ConnectingLane

Requirement **RS_ARSM_21**

For every ‘connectingLane’ (inside an instance of DF_Connection) the data element ‘maneuver’ shall be present.

Details:

Tested by:

Requirement **RS_ARSM_23**

The information in the data element ‘maneuver’ in ‘connectingLane’ shall be based on the lane marking arrows on the lane itself (if present).

In case there are no lane marking arrows on the street, the responsible human message designer shall decide the content of the data element individually for every intersection.

Details:

Tested by:

Requirement **RS_ARSM_22**

For data element ‘maneuver’ in ‘connectingLane’ exactly one of the first four bits of DE_AllowedManeuvers (i.e. exactly one direction indication per connectingLane) shall be set.

Details:

Tested by:

Requirement **RS_ARSM_24**

The maneuver indication ‘maneuverleft-/maneuverRightTurnonRedAllowed’ and ‘maneuverLaneChangeAllowed’ shall not be used.

Note: All other bits of the DE_AllowedManeuvers, which are not covered by this requirement or RS_ARSM_22 may be set but will not be used by current vehicle implementations.

Details:

Tested by:

6.2 SPATEM Automotive Requirements

6.2.1 Transmission and system clocks

Other (informational) **RS_ARSM_83**

The following requirement on SPATEM apply in addition to the relevant standards ([TS 103 301], [ISO/TS 19091], [SAE J2735]).

Details:

Tested by:

Requirement **RS_ARSM_98**

The system clock of the sub system responsible for the computation of the time change details shall be accurate to *tSubSystemClockAccuracy* with regards to the time base.

Details:

Tested by:

Requirement **RS_ARSM_99**

The system clocks of all sub systems contributing to the information generation of the SPATEM content shall by synchronized such that the clock deviations between them do not exceed *tIntraSystemClockAccuracy*.

Note: This includes both the RSU and the traffic light controller.

Details:

Tested by:

Requirement **RS_ARSM_92**

SPATEMs shall be transmitted with a transmission frequency of *fSpatTransmissionFreq*. DCC regulations shall always be fulfilled.

Details:

Tested by:

6.2.2 IntersectionState

Requirement **RS_ARSM_68**

The data field ‘id’ (DF_IntersectionReferenceID) shall be identical to the appropriate id tuple of the corresponding MAPEM ‘IntersectionGeometry’.

Details:

Tested by:

Requirement **RS_ARSM_69**

For the data element ‘status’ (of type IntersectionStatusObject) only the status bits ‘fixedTimeOperation’ (5), ‘trafficDependentOperation’ (6), ‘standbyOperation’ (7), ‘failureMode’ (8) or ‘off’ (9) shall be used. All other bits shall always be set to zero.

Note: Vehicles will mostly rely on MovementPhaseState without consideration of the IntersectionStatusObject.

Details:

Tested by:

Requirement **RS_ARSM_70**

Exactly one of the status bits referred to in RS_ARSM_69 shall be set to 1.

Details:

Tested by:

Requirement **RS_ARSM_52**

The data element ‘moy’ (DE_MinuteOfTheYear) in IntersectionState shall be set to the time of information generation, that is the time when the ‘timeChangeDetails’ are determined.

Details:

Tested by:

Requirement **RS_ARSM_53**

If the data element ‘timeStamp’ (DE_DSecond) in IntersectionState is present, it shall be set to the time of information generation, i.e. the time when the ‘timeChangeDetails’ are determined.

Details:

Tested by:

6.2.3 MovementList

Requirement **RS_ARSM_75**

Every given ‘signalGroup’ / ‘intersectionReferenceID’ tuple in the SPATEM shall be found in the MAPEM and vice versa.

Details:

Tested by:

Requirement **RS_ARSM_71**

The ‘states’ (DF_MovementList) shall be given at least for all connections through the intersection area with operational traffic lights (see definition of *‘operational’* (RS_ARSM_109), if the intersection status is either ‘fixedTimeOperation’ (5) or ‘trafficDependentOperation’ (6).

Details:

Tested by:

Requirement **RS_ARSM_80**

If a failure of the traffic light controller is detected (i.e. the IntersectionStatusObject indicates ‘failureMode’), either a SPaTEM with the eventState ‘0’ (unavailable) should be sent or SPATEM transmissions deactivated completely within less than *tDelayFailureTransmission* after the traffic light goes into failure mode.

Details:

Tested by:

Requirement **RS_ARSM_89**

An IntersectionState instance in SPATEM should not include duplicate MovementState instances in MovementList which over time only differ in the assigned SignalGroupID.

Note 1: Depending on the operation mode it is possible that in certain hours of the day two different MovementState instances (SignalGroups) have identical states. Therefore, this requirement is only stated as ‘should’.

Note 2: This implies that multiple lanes in MAPEM may observe the same SignalGroupID, in case the exact same movement rules apply to them at all times.

Details:

Tested by:

Requirement **RS_ARSM_74**

In case of multiple signals applying to one connection (e.g. for right turns) one singular virtual signal group with corresponding MovementState shall be transmitted, which reflects the combined MovementPhaseState of all applicable signals.

Details:

Tested by:

Requirement **RS_ARSM_78**

All events in ‘state-time-speed’ shall be sorted in chronological order with respect to *tAbsMinEndTime*.

Details:

Tested by:

Requirement **RS_ARSM_79**

At least MovementEvent instances for the current and next phase and all transitions in between shall be included in ‘state-time-speed’ (DF_MovementEventList).

Additional MovementEvent instances may be included.

Note: This means that the current and the next phase have to be included. If there is a transition in-between, three MovementEvent instances in total have to be included in the SPATEM.

Details:

Tested by:

Requirement **RS_ARSM_76**

The data element ‘eventState’ (of type DE_MovementPhaseState) shall represent the actual movement permissions according to the applicable traffic rules as indicated by the traffic lights (see also RS_ARSM_71 and RS_ARSM_74).

Note: The cars needs to know the applicable rules and not the physical representation / color of the physical traffic lights.

Details:

Tested by:

Requirement **RS_ARSM_72**

The MovementPhaseState ‘dark’ shall not be used.

Note 1: For the vehicle the applicable traffic rules are of relevance – not the physical representation. If no information can be given, ‘unavailable’ shall be used rather than ‘dark’.

Note 2: According to the other requirements on MovementPhaseState (i.e. RS_ARSM_74 and RS_ARSM_76), there is no situation left in which ‘dark’ needs to be used.

Details:

Tested by:

Requirement **RS_ARSM_77**

The data element ‘eventState’ shall be set to the applicable value considering the distinction between protected and permissive movements.

Details:

Tested by:

Requirement **RS_ARSM_103**

The MovementPhaseState ‘stop-And-Remain’ shall be used when vehicles on corresponding lanes are not allowed to enter the conflict zone.

Note: In most cases, this corresponds to the traffic light showing ‘red’.

Details:

Tested by:

Requirement **RS_ARSM_104**

The MovementPhaseState ‘pre-Movement’ shall be used for transitions that directly precede the phase ‘permissive-Movement-Allowed’ or ‘protected-Movement-Allowed’.

Note: For example in Germany, this corresponds to the traffic light showing ‘red-yellow’.

Details:

Tested by:

Requirement **RS_ARSM_105**

The MovementPhaseState ‘permissive-Movement-Allowed’ shall be used when vehicles on corresponding lanes are allowed to enter the conflict zone but there still might occur conflicting traffic which they have to pay attention for.

Note: This applies for example in some right-turn situations when the driver needs to pay attention to pedestrians which might cross the street because they also are allowed to enter the conflict zone.

Details:

Tested by:

Requirement **RS_ARSM_106**

The MovementPhaseState ‘protected-Movement-Allowed’ shall be used when vehicles on corresponding lanes are allowed to enter the conflict zone and there shouldn’t be any conflicting traffic according to the traffic rules.

Note: This applies for example in some left-turn situations when only lanes having a left-turn connections are in a ‘Movement-Allowed’ state but no other conflicting traffic.

Details:

Tested by:

Requirement **RS_ARSM_107**

The MovementPhaseState ‘permissive-Clearance’ shall be used when vehicles on corresponding lanes

- are allowed to enter the conflict zone if they are not able to stop before the stop line
- shall clear the conflict zone
- and have to be attentive of potential conflicting traffic.

Note: In Germany this corresponds to the traffic light showing ‘yellow’.

Details:

Tested by:

Requirement **RS_ARSM_110**

The MovementPhaseState ‘protected-Clearance’ shall be used when vehicles on corresponding lanes

- are allowed to enter the conflict zone if they are not able to stop before the stop line,
- shall clear the conflict zone and
- there shouldn’t be any conflicting traffic according to the traffic rules.

Note: In Germany this corresponds to the traffic light showing ‘yellow’.

Details:

Tested by:

Requirement **RS_ARSM_108**

The MovementPhaseStates ‘caution-Conflicting-Traffic’ shall be used for signalGroups belonging to lanes of minor roads if none of the aforementioned MovementPhaseStates are applicable (e.g. if the traffic light controller is in standby mode). It shall indicate that vehicles are allowed to proceed but have to give way to conflicting traffic.

Note: In Germany this corresponds to the traffic light showing ‘flashing yellow’.

Details:

Tested by:

6.2.4 TimeChangeDetails

Requirement **RS_ARSM_120**

The data field ‘timing’ (of type TimeChangeDetails) shall be present for every instance of MovementEvent in SPATEM that precedes a MovementEvent instance representing a phase (i.e. containing an instance of MovementPhaseState having one of the values 2, 3, 5 or 6).

Note: See also RS_ARSM_78 and RS_ARSM_79.

Details:

Tested by:

Other (informative) **RS_ARSM_54**

Data elements of type TimeMark (i.e. ‘startTime’, ‘minEndTime’, ‘maxEndTime’, ‘likelyTime’, ‘nextTime’) shall represent 1/10 s in the hour in which the state change may occur (this may be the hour represented by the entry ‘moy’ or the following hour).

Note: If for the received TimeMark it holds that

*TimeMark / 10 s < (moy modulo 60) min * 60 s/min*,

the TimeMark corresponds to the hour following the hour represented by ‘moy’.

Details:

Tested by:

Requirement **RS_ARSM_56**

The data element ‘minEndTime’ shall have a value *between pTimeMarkMin* and *pTimeMarkOutOfRange*.

Note: This means that the value *pTimeMarkUnknown* (unknown) shall not be used.

Details:

Tested by:

Requirement **RS_ARSM_55**

The data element ‘minEndTime’ (DE_TimeMark) shall be set for every signal group to the earliest time possible at which the phase state of the respective signal group could change, including unpredictable events like pedestrian crossing or pre-emption for emergency and other priority vehicles (e.g. public transport). The risks of force majeure such as technical failures shall not be considered in the determination of ‘minEndTime’.

Note: That means the minEndTime may be the currentTime + the time it takes to change the signal if a prioritization request occurred at the current time.

Details:

Tested by:

Requirement **RS_ARSM_91**

In successive SPATEM transmissions, the instant, which the 'minEndTime' of one MovementState refers to, shall not move to an earlier point in time. It may however progress to a later point in time.

Note: In relative terms this means that the remaining time until 'minEndTime' shall not decrease faster than the time passes.

Details:

Tested by:

Requirement **RS_ARSM_57**

The data element ‘maxEndTime’ (DE_TimeMark) shall be present for actuated traffic light operation.

Details:

Tested by:

Requirement **RS_ARSM_58**

The data element ‘maxEndTime’ (DE_TimeMark) shall be set to the latest time possible at which the phase state could change.

Details:

Tested by:

Requirement **RS_ARSM_59**

In case ‘maxEndTime’ is infinite (e.g. for traffic lights that only change in case of pedestrian requests), the value shall be set to *pTimeMarkOutOfRange*.

Note: This includes the case when the actual maxEndTime is not known.

Details:

Tested by:

Requirement **RS_ARSM_60**

For ‘maxEndTime’ the value *pTimeMarkUnknown* (unknown) shall not be used.

Details:

Tested by:

Requirement **RS_ARSM_90**

The instant, which 'maxEndTime' refers to, shall not progress to a later point in time. It may however move to an earlier point in time.

Note: In relative terms this means that the remaining time until 'maxEndTime' shall not increase.

Details:

Tested by:

Requirement **RS_ARSM_61**

For traffic signal controllers operating fixed time, where the time of change is known, ‘minEndTime‘, ‘likelyTime’ and ‘maxEndTime’ shall be equal, if they are present.

Details:

Tested by:

Requirement **RS_ARSM_62**

For traffic signal controllers operating fixed time, where the time of change is known, ‘minEndTime’ shall be accurate to the displayed change of the traffic light within *tTimeOfChangeAccuracy*.

[Note: For tests the electrical controller output to the lights may be measured.]

Details:

Tested by:

Other (informational) **RS_ARSM_63**

The ‘likelyTime’ shall be used to convey the most likely time the phase state changes.

Note: RS_ARSM_102 states how to use the confidence data element.

Details:

Tested by:

Requirement **RS_ARSM_64**

The data element likelyTime (DE_TimeMark) shall be present for actuated traffic light operation.

Note: The confidence for the likelyTime is given in the DE ‘confidence’ (DE_TimeIntervalConfidence).

Details:

Tested by:

Other (informational) **RS_ARSM_65**

Let tAbsMinEndTime, tAbsLikelyTime and tAbsMaxEndTime be the instants which 'likelyTime', 'minEndTime' and 'maxEndTime refer to, they suffice the following condition:

tAbsMinEndTime <= tAbsLikelyTime <= tAbsMaxEndTime.

Details:

Tested by:

Requirement **RS_ARSM_66**

For the data element 'likelyTime' the value *pTimeMarkUnknown* (unknown) shall not be used.

Details:

Tested by:

Requirement **RS_ARSM_115**

If the data element 'likelyTime' is present, the confidence of 'likelyTime' shall be present as well.

Details:

Tested by:

Other (informational) **RS_ARSM_101**

The given probabilities for the TimeIntervalConfidence apply for the discrete time of the likelyTime, assuming the signal group state changes at whole second intervals. Otherwise, if the signal group state change is not assumed to be at a discrete second, the time interval around the likelyTime, for which the given probabilities apply, is defined as ± tTimeChangeInterval.

Note 1: Normally the cycle time of the traffic light controller is one second. Therefore, the intervals between state changes can only be multiple of whole seconds.

Note 2: Actually, this definition would always be correct. Even in the first case, if the state change is at discrete seconds. Therefore, this rule could always be applied.

Details:

Tested by:

Requirement **RS_ARSM_102**

The data element TimeIntervalConfidence shall be used as defined in SAE J2735. Confidence shall be interpreted as probability that the real phase change occurs within ± *tTimeChangeInterval* of the indicated likelyTime.

Note 1: This means that the probability for likelyTime – tTimeChangeInterval <= phase change time <= likelyTime + *tTimeChangeInterval* shall be indicated.

Note 2: Implementation of one confidence threshold for all situations on receiving side will not work. It is recommended to evaluate the confidence in relation to the prediction horizon with different thresholds for the different use cases.

Details:

Tested by:

Requirement **RS_ARSM_67**

If no prediction is available, the confidence of 'likelyTime' shall be disseminated with the value '0'.

Details:

Tested by:

---

7 Annex

This annex contains tables for MAPEM and SPATEM showing which data elements are mandatory according to the standards (SAE, CEN/ISO), this document and the C-Roads profile in Release 1.6.

Legend:

- The number of '+' in the column 'Layer' and the shading of the row represents the layer / level of the corresponding data element within the message.
- '-': This data element is not mentioned in the respective document.
- 'O': This data element is optional.
- 'M': This data element is mandatory.
- 'O/M': This data element is mandatory only under certain conditions which are defined in the respective document.
- 'C': This data element is an option within a 'Choice'.
- 'NU': (C-Roads specific) This data element is not used in C-Roads.
- 'F': The respective document forbids the usage of this data element.

7.1 MAPEM mandatory and optional data elements

Layer	Data element / data field in MapData	Standards	C2C-CC (this document)	C-Roads (Release 1.6)	Combined
+	timestamp	O	O	O	O
+	msgIssueRevision	M	-	M	M
+	layerType	O	-	O	O
+	layerID	O	-	O	O
+	intersections (list of IntersectionGeometry)	O	-	M	M
++	name	O	O	O	O
++	id	M	M	M	M
+++	region	O	M	O	M
+++	id	M	M	M	M
++	revision	M	-	M	M
++	refPoint	M	M	M	M
+++	…	M	-	M	M
++	laneWidth	O	M	O	M

++	speedLimits (list of speedLimitType)	O	-	O	O
++	laneSet (list of GenericLane)	M	M	M	M
+++	laneID	M	M	M	M
+++	name	O	O	O	O
+++	ingressApproach	O	O/M	O	O/M
+++	egressApproach	O	O/M	O	O/M
+++	laneAttributes	M	-	M	M
++++	directionalUse	M	-	M	M
++++	sharedWith	M	-	M	M
++++	laneType (Choice)	M	M	M	M
+++++	vehicle Lane	C	-	C	C
+++++	crosswalkLane	C	-	C	C
+++++	Bikelane	C	-	C	C
+++++	Sidewalk	C	-	NU	
+++++	medianLane	C	-	NU	
+++++	stripingLane	C	-	NU	
+++++	trackedVehicle	C	-	C	C
+++++	parkingLane	C	-	NU	
+++	maneuvers	O	F	NU	F
+++	nodeListXY (Choice)	M	M	M	M
++++	nodes (list of NodeXY)	C	M	M	M
+++++	delta (Choice)	M	M	M	M
++++++	…		-		
+++++	attributes	O	O/M	O	O/M
++++++	localNode (list of NoteAttributeXY)	O	O/M	O	O/M
++++++	enabled (list of SegmentAttributeXY)	O	O/M	O	O/M

++++++	disabled (list of SegmentAttributeXY)	O	O/M	O	O/M
++++++	data	O	-	O	O
+++++++	…				
++++++	dWidth	O	-	O	O
++++++	dElevation	O	-	O	O
++++++	regional	O	-	O	O
++++	computed	C	F	F	F
+++	connectsTo (list of Connections)	O		O	
++++	connectingLane	M	M	M	M
+++++	lane	M	M	M	M
+++++	maneuver	O	M	O	M
++++	remoteIntersection	O	-	O	O
+++++	…		-		
++++	signalGroup	O/M	O/M	O	O/M
++++	userClass	O	-	O	
++++	connectionID	O	-	M	M
+++	overlays	O	-	NU	
+++	regional	O	-	O	O
++	preemptPriorityData	O	-	O	O
++	regional	O	-	O	O
+	roadSegments	O	-	O	O
+	dataParameters	O	-	O	O
+	restrictionList	O	-	O	O
+	regional	O	-	O	O

7.2 SPATEM mandatory and optional data elements

Layer	Data element / data field in SpatData	Standards	C2C-CC (this document)	C-Roads (Release 1.6)	Combined
+	timestamp	O	-	O	O
+	name	O	-	O	O
+	intersections (list of IntersectionState)	M	M	M	M
++	name	O	-	O	
++	id	M	M	M	M
+++	region	O	M	O	M
+++	id	M	M	M	M
++	revision	M	-	M	M
++	status	M	M	M	M
++	moy	O	M	M	M
++	timestamp	O	O	M	M
++	enabledList (list of LaneID)	O	-	O/M	O/M
++	states (list of MovementState)	M	M	M	M
+++	movementName	O	-	O	O
+++	signalGroup	M	M	M	M
+++	state-time-speed (list of MovementEvent)	M	M	M	M
++++	eventState	M	M	M	M
++++	timing	O	O/M	O (O/M)	O/M
+++++	startTime	O	-	O	O
+++++	minEndTime	M	M	M	M
+++++	maxEndTime	O	O/M	M	M
+++++	likelyTime	O	O/M	O	O/M
+++++	confidence	O	O/M	O/M	O/M

+++++	nextTime	O	-	O	O
++++	speeds (list of AdvisorySpeed	O	-	O	O
+++++	…				
++++	regional	O	-	O	O
+++	maneuverAssistList (list of ConnectionManeuverAssist)	O	-	O	O
++++	…				
+++	regional	O	-	O	O
++	maneuverAssistList (list of ConnectionManeuverAssist)	O	-	O	O
+++	…				
++	regional	O	-	O	O
+	regional	O	-	O	O