



monotch

SMART MOBILITY PLATFORMS

TLEX-TLC-SYSTEM-VLOG Interface v1.4.1

Table of contents

1	Versioning.....	3
2	Referenced documents	5
3	Introduction.....	6
4	Overview	7
5	API.....	8
5.1	Authentication and authorization	8
5.2	API endpoints	8
5.2.1	Sessions.....	9
6	VLOG protocol.....	15
6.1	Connection establishment.....	15
6.2	MAP from ITF	16
6.2.1	MAP field mapping	16
6.3	SPAT from VLOG	22
6.3.1	TR based clock difference detection	23
6.3.2	SPAT trigger	23
6.3.3	WPS changes.....	23
6.3.4	MovementState generation	24
6.3.5	SPAT field mapping	28
6.3.6	SPAT payload merging	33

1 Versioning

This document is using a versioning scheme that indicates the version of this TLEX interface and tracks the revisions of this document. This version scheme is <interface version major>.<interface version minor>.<document revision>. The first two version numbers (major and minor) indicate the version of the interface and only change when there is technical change in the described interface. Major version is only bumped when there is compatibility breaking change. Minor version is bumped on trivial, non breaking changes of the interface. The last version number indicates the revision of this document.

Version	Date	Author	Changes
1.0.0	25 Jul 2017	L. Rijnveld	Initial specification for TLEX release v1.1
1.1.0	29 Aug 2017	Y. Duran	<p>Updated for TLEX release v1.2:</p> <p>Chapter: "VLOG Protocol" updated:</p> <ol style="list-style-type: none"> 1. ITF specification v1.2 (was v0.97); 2. MAP specification v1.2 (was v1.0); 3. SPAT specification v1.2 (was v1.0); 4. Removed LayerType and LayerId; 5. Added DataParams and msgIssueRevision; 6. Added GenericLane.regional (connectionTrajectory); 7. Updated chapter "WPS triggered SPAT" with MovementEvent state mappings; 8. Added Spat and Map header.stationID generation; 9. Added Spat enabledLanes; <p>Chapter: "ITF XSD" added.</p>
1.2.0	14 Mar 2018	L. Rijnveld	<p>Updated for TLEX release v1.3:</p> <p>Chapter: "VLOG Protocol": Updated SPAT Mapping 4.5 "TimeChangeDetails.confidence" conversion specification</p>

Version	Date	Author	Changes
1.3.0	26 Apr 2019	L. Rijneveld	<p>Updated for TLEX release v1.4:</p> <p>Chapter: "VLOG Protocol": rewritten due to the following specification changes:</p> <ol style="list-style-type: none"> 1. VLOG specification v3.0.1, document v2.6.2 (was v3.0.0, document v2.5.2); 2. ITF specification v2.1 (was v1.2); 3. SPAT specification v2.0 (was v1.2); 4. Clock difference detection; 5. SPAT triggered on RC messages instead of WPS or FT messages; 6. SPAT always contains a MovementState for each SignalGroup found in the ITF ControlData SignalGroup list; 7. MovementState's first MovementEvent contains WR message based "stateChangeReason" if applicable; 8. MovementEvent timing field "startTime" always empty; 9. MovementEvent timing field "confidence" based on VLOG percentage mapping; 10. SPAT no longer suppressed when intersectionStatus indicates non-operational TLC. <p>Chapter: "ITF XSD" removed (no longer needed since default v2.1 XSD is used).</p>
1.4.0	22 Nov 2018	L. Rijneveld	<p>Updated for TLEX release v1.7:</p> <p>Chapter: "VLOG Protocol": ITF to MAP conversion changes for the following fields:</p> <ul style="list-style-type: none"> • 4.1 - RegulatorySpeedLimit.type • 4.2 - RegulatorySpeedLimit.speed • 5.10 - GenericLane.regional • 7.2 - NodeXY.attributes • 11.2 - RestrictionUserType.regional
1.4.1	23 Mar 2020	L. Rijneveld	Formatting and layout improvements

2 Referenced documents

ID	Reference	Version	Date
[1]	TLEX-TLC-ADMIN Interface	1.2.1	23 Mar 2020
[2]	V-Log protocol en definities	2.6.2	01 Nov 2017
[3]	Intersection Topology Format (ITF) PROFILE	2.1	22 Mar 2018
[4]	MapData (MAP) PROFILE	1.2	29 Jun 2017
[5]	SPAT PROFILE	2.0	16 Nov 2017

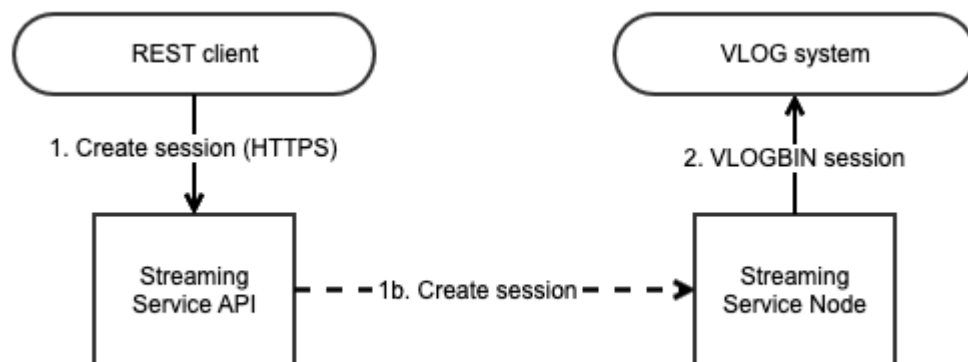
3 Introduction

This document describes the technical interface with TLEX for TLC systems using the VLOG protocol. The primary goal of this document is to track/version the interface specification for this explicit context so that the impact of future TLEX specification changes can be properly assessed.


4 Overview

VLOG systems interface with TLEX on two levels:

1. JSON-REST API: used for requesting/creating VLOG sessions;
2. VLOG protocol: VLOG protocol used for sending VLOG datagrams to TLEX which are converted into SPAT.



The JSON-REST API call is only needed when there is a need for establishing the VLOG connection manually. It is possible to have automatic VLOG connection establishment enabled within TLEX; however this is managed through the TLEX-TLC-ADMIN Interface (see referenced document [1]), and therefore not in scope of this document.

 The VLOG protocol itself is not part of this specification. TLEX VLOG sessions convert ITF documents to MAP and VLOG datagrams to SPAT. The specification of this conversion is described in this document.

5 API

5.1 Authentication and authorization

The authentication of the Client will be based on a "authorization token". This "authorization token" needs to be passed as the "X-Authorization" request header value. The authorization token needs to belong to an "TLC_SYSTEM" authorization (for more details the "authorization model" chapter in the TLEX-TLC-ADMIN Interface (see referenced document [1])).

5.2 API endpoints

API endpoint			
End point	Method	URI	Description
sessions	POST	/sessions	Creates a new streaming session

5.2.1 Sessions

5.2.1.1 POST /sessions

Creates a new streaming session.

5.2.1.1.1 Request

```
POST <API base URL>/sessions HTTP/1.1
Host: <hostname>
X-Authorization: <authorization token>
Content-Type: application/json
```

```
{
  "domain": "<domain>",
  "type": "<type>",
  "protocol": "<protocol>",
  "details": {
    <protocol details>
  }
}
```

Name	Description
domain	Sessions are created within a specific domain, identified by a single string Only sessions created for the same domain will be able to stream data to each other
type	The session type must be "TLC".
protocol	The session protocol must be "VLOG".
details	Session protocol specific details for creating the session

5.2.1.1.2 Response

HTTP/1.1 200 OK
 Content-Type: application/json

```
{
  "token": "<token>",
  "domain": "<domain>",
  "type": "<type>",
  "protocol": "<protocol>",
  "details": {
    <protocol details>
  }
}
```

Name	Description
token	The token for the created session
domain	See request
type	See request
protocol	See request
details	Session protocol specific details of the created session

5.2.1.1.3 Session type "TLC" with protocol "VLOG"

VLOG (ASCII) based streaming session for one specific TLC.

VLOG sent by the Client will be converted to SPAT payloads and received by "Broker" session Clients if the "TLC identifier" of this specific TLC is within their scope.

Since the VLOG protocol is uni-directional, the payloads sent by "Broker" session clients having a payload "TLC identifier" that matches this specific TLC's identifier will be ignored.

5.2.1.1.3.1 Request details

```
{
  "tlcIdentifier": "<TLC identifier>",
  "host": "<host address>",
  "port": <TCP port number>,
  "itf": "<Base64 encoded ITF XML document>"
}
```

Name	Description
tlcIdentifier	The TLC identifier for the session
host	The host address on which the VLOG ASCII device will accept the TCP connection
port	The TCP port on which the VLOG ASCII device will accept the TCP connection
itf	<p>A base64 encoded ITF XML document. The ITF must meet the following criteria:</p> <ul style="list-style-type: none"> Valid XML, based on ITF XSD; Must contain at least one SignalGroup definition in the ControlData section; MapData must be MAP compliant.

5.2.1.1.3.2 Response details

```
{
  "remote": {
    "host": "<host address>",
    "port": <port number>,
    "expiration": "<ISO 8601 date time>"
  },
  "tlcIdentifier": "<TLC identifier>",
  "itf": "<Base64 encoded ITF XML document>",
  "payloadRateLimit": <payload/second limit>,
  "payloadRateLimitDuration": "<ISO 8601 duration>",
  "payloadThroughputLimit": <KB/second limit>,
  "payloadThroughputLimitDuration": "<ISO 8601 duration>"
}
```


Name	Description
tlcIdentifier	See request details
remote	The remote VLOG device details for establishing the TCP connection
remote.host	The host address of the VLOG device
remote.port	The TCP port of the VLOG device
remote.expiration	<p>The expiration date time of the listener in ISO 8601 date time format</p> <p>If the TCP connection has not been established before this time the connect attempt will expire and the streaming session will end</p> <p>The default connect expiration will be set to 10 seconds</p>
payloadRateLimit	<p>The maximum amount of payloads per second allowed for the streaming session</p> <p>If the average amount of received payloads per second during the duration (see payloadRateLimitDuration) exceeds the limit the Streaming Service will terminate the TCP connection</p>
payloadRateLimitDuration	The period, in ISO 8601 duration format, during which the average amount of received payloads per second should not exceed the payloadRateLimit
payloadThroughputLimit	<p>The maximum amount of payload kilobytes (KB) per second allowed for the streaming session</p> <p>If the average amount of received payload kilobytes (KB) per second during the duration (see payloadThroughputLimitDuration) exceeds the limit the Streaming Service will terminate the TCP connection</p>

Name	Description
payloadThroughputLimitDuration	The period, in ISO 8601 duration format, during which the average amount of received payload kilobytes (KB) per second should not exceed the payloadThroughputLimit

5.2.1.1.3.3 Example


```
POST api/v1/sessions HTTP/1.1
Host: api.tlex.eu
X-Authorization: dtNB_vhvJ0wgTGf1N0DxN38_AmTL_4yiPRZdqZSuK3k
Content-Type: application/json

{
  "domain": "test",
  "type": "TLC",
  "protocol": "VLOG",
  "details": {
    "tlcIdentifier": "NLZH0024",
    "host": "vri.foo.bar",
    "port": 7070,
    "itf": "77u/PD94bwwgdmVyc2lvcj0iMS4wIiBlb ... YT4NCjwvdG9wb2xvZ3k+"
  }
}
```

 For the sake of readability, the "itf" field's value has been clipped.

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "token": "dtNB_vhvJ0wgTGf1N0DxN38_AmTL_4yiPRZdqZSuK3k",
  "domain": "test",
  "type": "TLC",
  "protocol": "VLOG",
  "details": {
    "remote": {
      "host": "vri.foo.bar",
      "port": 7070,
      "expiration": "2016-11-17T16:04:23Z"
    },
    "tlcIdentifier": "NLZH0024",
    "itf": "77u/PD94bwwgdmVyc2lvcj0iMS4wIiBlb ... YT4NCjwvdG9wb2xvZ3k+",
    "payloadRateLimit": 45,
    "payloadRateLimitDuration": "PT5S",
    "payloadThroughputLimit": 45,
    "payloadThroughputLimitDuration": "PT5S"
  }
}
```

 For the sake of readability, the "itf" field's value has been clipped.

6 VLOG protocol

The VLOG protocol is used for TLC's that do not implement the TCPStreaming protocol and facilitates a continuous uni-directional stream of VLOG ASCII messages as described in the VLOG specification document "V-Log protocol en definitives v2.6.2" dated 01-11-2017.

For sessions running the VLOG protocol the Streaming Service is not payload agnostic but will interpret the VLOG ASCII messages in order to produce SPAT payloads. Since SPAT payloads are useless for the receiver without a valid MAP payload the Streaming Service also produces the MAP payload by generating MAP from ITF.

6.1 Connection establishment

When the VLOG protocol is used, the Streaming Service will establish a TCP connection to the specified host/port combination (see API end-point "sessions"). Even though the VLOG specification document describes a CCOL V-Log command "VLOGASCII" for starting a VLOG ASCII dump, the Streaming Service expects the host to start streaming VLOG ASCII as soon as the connection is established and will not sent any CCOL V-Log command.

6.2 MAP from ITF

As soon as the connection is established with the VLOG device the Streaming Service will convert the specified ITF definition (see API end-point "sessions" and/or API end-point "tlcs" regarding providing the ITF) into a MAP payload.

The ITF specification version used for MAP generation is v2.1 as described in the Dutch ITF Profile document "Intersection Topology Format (ITF) PROFILE v2.1" dated 22-03-2018.

6.2.1 MAP field mapping

The following table specifies the MAP field mapping in relation to the ITF definition. It uses the following convention regarding the "mapping type" to illustrate the type of data mapping:


- static: the field will always be set by the listed "hardcoded" value;
- transparent: the field will be set with the exact same value as defined in the ITF.
- converted: the field's value is derived/converted from the value(s) in the ITF.

The MAP level numbering refers to the MAP specification as described in the Dutch MAP Profile document "MapData (MAP) PROFILE v1.2" dated 29-06-2017.

MAP Level	MAP Field	Mapping type	Mapping/Value
MAPEM.1.1	header.protocolVersion	static	1
MAPEM.1.2	header.messageID	static	5
MAPEM.1.3	header.stationID	converted	Formula: (IntersectionGeometry.id.region * 65536) + IntersectionGeometry.id.id The region and id are from the 1st IntersectionGeometry in the list. The id's last digit is changed to 0.
MAPEM.2	map	<Level 0>	
Level 0: MapData			
0.1	timeStamp	static	<empty>
0.2	msgIssueRevision	transparent	<ITF>
0.3	layerType	static	<empty>
0.4	layerID	static	<empty>

0.5	intersections	<Level 1>	
0.6	roadSegments	static	<empty>
0.7.1	dataParameters.processMethod	static	<empty>
0.7.2	dataParameters.processAgency	transparent	<ITF>
0.7.3	dataParameters.lastCheckedDate	transparent	<ITF>
0.7.4	dataParameters.geoidUsed	static	<empty>
0.8	restrictionList	<Level 3>	
0.9	regional	static	<empty>
Level 1: IntersectionGeometry			
1.1	name	transparent	<ITF>
1.2.1	id.region	transparent	<ITF>
1.2.2	id.id	transparent	<ITF>
1.3	revision	transparent	<ITF>
1.4	refPoint	<Level 12>	
1.5	laneWidth	transparent	<ITF>
1.6	speedLimits	<Level 4>	
1.7	laneSet	<Level 5>	
1.8	preemptPriorityData	static	<empty>
1.9	regional	static	<empty>
Level 2: RoadSegment			
Not used			

Level 3: RestrictionClassAssignment			
3.1	id	transparent	<ITF>
3.2	users	<Level 11>	
Level 4: RegulatorySpeedLimit			
4.1	type	transparent	<ITF> <div>⚠ When type == "nominalSpeed" the "RegulatorySpeedLimit" entry is ignored</div>
4.2	speed	transparent	<ITF> <div>⚠ When type == "nominalSpeed" the "RegulatorySpeedLimit" entry is ignored</div>
Level 5: GenericLane			
5.1	laneID	transparent	<ITF>
5.2	name	transparent	<ITF>
5.3	ingressApproach	transparent	<ITF>
5.4	egressApproach	transparent	<ITF>
5.5.1	laneAttributes.directionalUse	transparent	<ITF>
5.5.2	laneAttributes.sharedWith	transparent	<ITF>
5.5.3	laneAttributes.laneType	<Level 6>	
5.5.4	laneAttributes.regional	static	<empty>
5.6	maneuvers	static	<empty>
5.7.1	nodeList.nodes	<Level 7>	

5.7.2	nodeList.computed	static	<empty>
5.8	connectsTo	<Level 9>	
5.9	overlays	static	<empty>
5.10	regional (connectionTrajectory)	static	<empty>
Level 6: LaneTypeAttributes			
6.1	vehicle	transparent	<ITF>
6.2	crosswalk	transparent	<ITF>
6.3	bikelane	transparent	<ITF>
6.4	sidewalk	static	<empty>
6.5	median	static	<empty>
6.6	striping	static	<empty>
6.7	trackedVehicle	transparent	<ITF>
6.8	parking	static	<empty>
Level 7: NodeXY			
7.1	delta	converted	<ITF> Each node-LatLon is converted to a NodeOffsetPointXY where <ul style="list-style-type: none"> • The first node is calculated relative the IntersectionGeometry.refPoint • Subsequent nodes are calculated relative the previous node
7.2	attributes	transparent	<ITF> <div>  NodeAttributeXY "yield" is ignored </div>

Level 8: ComputedLane			
Not used			
Level 9: Connection			
9.1.1	connectingLane.laneID	transparent	<ITF>
9.1.2	connectingLane.maneuver	transparent	<ITF>
9.2.1	remoteIntersection.region	transparent	<ITF>
9.2.2	remoteIntersection.id	transparent	<ITF>
9.3	signalGroup	transparent	<ITF>
9.4	userClass	transparent	<ITF>
9.5	connectionID	transparent	<ITF>
Level 10: LaneDataAttribute			
10.1	pathEndPointAngle	static	<empty>
10.2	laneCrownPointCenter	static	<empty>
10.3	laneCrownPointLeft	static	<empty>
10.4	laneCrownPointRight	static	<empty>
10.5	laneAngle	static	<empty>
10.6	speedLimits	<Level 4>	
10.7	regional	static	<empty>
Level 11: RestrictionUserType			
11.1	basicType	transparent	<ITF>

11.2	regional (emissionType)	static	<empty>
Level 12: Position3D			
12.1	lat	transparent	<ITF>
12.2	long	transparent	<ITF>
12.3	elevation	static	<empty>
12.4	regional	static	<empty>

6.3 SPAT from VLOG

After sending the generated MAP payload, the Streaming Service will start parsing the incoming VLOG ASCII data stream. Although the VLOG protocol describes a vast variety of message types, only 15 message types are relevant for SPAT generation:

Abbreviation	VLOG message type	Description
TR	1	Time reference status message Contains the reference time for timestamp calculations
DP	5, 6	Detector status message Contains the state of a detection input
IS	7, 8	Input status message Contains the state of a "other" input
US	11, 12	Output status message Contains the state of a "other" output
FC	13, 14	Phase-cycle actual status message Contains the actual output state (WUS) of a signal group output
WPS	19, 20	Actual program state message Contains the current program state
FT	36	Phase-cycle timing message Contains one or more signal group state timings
WR	37, 38	Wait reason Contains one or more signal group wait reasons
RC	128	Realtime Check Message Contains CRC code over the data since the previous RC message

Other VLOG message types will be ignored.

The TR messages are used to determine the VLOG time reference.

The RC messages are used to trigger SPAT generation.

The FT, WPS and WR messages are used to determine the state of the intersection and it's signal groups (see SPAT field mapping 1.4 and 1.8).

The DP, IS, US and FC messages are used to determine the active variant / enabled lanes (see SPAT field mapping 1.7).

Since it is mandatory for a SPAT message to convey the state of all signal groups in each SPAT message, the VLOG to SPAT process will keep a register of last known/received WPS message and last known/received FT and WR messages for each signal group.

6.3.1 TR based clock difference detection

It is important that all connected systems have their times properly synced to a reliable clock source. VLOG uses the TR (Time Reference) message to convey the time reference for following VLOG messages. Since only live/streaming VLOG should be handled for SPAT conversion, the VLOG time reference should never exceed 3 seconds of difference compared to the Streaming Service system clock. If a TR message is received that exceeds this threshold the session will be terminated.

6.3.2 SPAT trigger

SPAT is generated upon the reception of a RC message, but only after the reception of at least one TR message.

6.3.3 WPS changes

When a WPS status or change message is received which contains a different WPS state, the VLOG message register will be re-initialized. Upon re-initilisation all last known/received FT and WR messages will be cleared from the register.

The default WPS status on session initialisation will be set to "Regelen". This will prevent register re-initilisation in the common case where a WPS program status "Regelen" is received in the beginning of a new VLOG session.

6.3.4 MovementState generation

During SPAT generation, a MovementState will be generated for each SignalGroup found in the ITF ControlData SignalGroup list.

6.3.4.1 WPS based MovementState

When no last known/received FT message is present in the message register for a given SignalGroup a MovementEventList with only one MovementEvent will be generated based on the last known or default WPS program status:

WPS program status	MovementEvent.eventState
Ongedefineerd	unavailable
Gedoofd	dark
Geel knipperend	caution_Conflicting_Traffic
Statisch geel	permissive_clearance
Alles rood	stop_And_Remain
Regelen	unavailable

The MovementEvent will not contain any timing information (MovementEvent.timing will be left empty).

6.3.4.2 FT based MovementState

When a last known/received FT message is present in the message register the MovementEventList will be derived from this FT message. The MovementEventList will contain a MovementEvent for each FT "event" in the same order as present in the FT message:

VLOG FT event status	MovementEvent.eventState
0: "Onbekend"	unavailable
1: "Gedoofd"	dark
2: "Rood knipperend signaal"	stop-And-Remain
3: "Rood"	stop-And-Remain
4: "Groen overgang"	pre-Movement
5: "Groen met deelconflict"	permissive-Movement-Allowed
6: "Groen zonder deelconflict"	protected-Movement-Allowed
7: "Geel met deelconflict"	permissive-Clearance
8: "Geel zonder deelconflict"	protected-Clearance
9: "Geel knipperen"	caution-Conflicting-Traffic
10: "Groen knipperen deelconflict"	permissive-Clearance
11: "Groen knipperen"	protected-Clearance

6.3.4.2.1 Timing confidence


The MovementEvent's timing confidence will be derived from the FT event using the following mapping:

VLOG "betrouwbaarheid" (from)	VLOG "betrouwbaarheid" (thru)	SPAT confidence
0	21	0
22	36	1
37	47	2
48	56	3
57	62	4
63	68	5
69	73	6
74	77	7
78	81	8
82	85	9
86	88	10
89	91	11
92	94	12
95	96	13
97	98	14
99	100	15

6.3.4.3 MovementState stateChangeReason

The first MovementEvent in the list will contain the regional "stateChangeReason" if a prior WR message was received.

VLOG "reden wachttijd"	SPAT stateChangeReason	Priority
0: OV Ingriep	publicTransportPriority	3
1: Hulpdienst ingriep	emergencyVehiclePriority	1
2: Trein ingriep	trainPriority	4
3: Brug ingriep	bridgeOpen	5
4: Hoogtemelding	vehicleHeight	2
5: Weersingriep	weather	12
6: File ingriep	trafficJam	6
7: Tunnel afsluiting	tunnelClosure	11
8: Doseran actief	meteringActive	7
9: Vrachtverkeer ingriep	truckPriority	8
10: Fiets peloton ingriep	bicyclePlatoonPriority	9
11: Voertuig peloton ingriep	unknown	10

 VLOG could define multiple "reden wachttijd" at the same time. In this case the stateChangeReason with the highest priority (lowest number) will be used.

6.3.5 SPAT field mapping

The following table specifies the SPAT field mapping in relation to the generated MAP and received VLOG messages. It uses the following convention regarding the "mapping type" to illustrate the type of data mapping:


- static: the field will always be set by the listed "hardcoded" value;
- mapped: the field will be set by the referred MAP value;
- converted: the field's value is derived/converted from the value(s) in VLOG message(s).

The SPAT level numbering refers to the SPAT specification as described in the Dutch SPAT Profile document "SPAT PROFILE v2.0" dated 16-11-2017.

SPAT level	SPAT field	Mapping type	Mapping/Value
SPATEM.1.1	header.protocolVersion	static	1
SPATEM.1.2	header.messageID	static	4
SPATEM.1.3	header.stationID	converted	Formula: (IntersectionGeometry.id.region * 65536) + IntersectionGeometry.id.id The region (MAP 1.2.1) and id (MAP 1.2.2) are from the 1st IntersectionGeometry in the list. The id's last digit is changed to 0.
SPATEM.2	spat	<Level 0>	
Level 0: SPAT			
0.1	timeStamp	static	<empty>
0.2	name	static	<empty>
0.3	intersections	<Level 1>	
0.4	regional	static	<empty>
Level 1: IntersectionState			
1.1	name	mapped	MAP 1.1 (IntersectionGeometry.name)

1.2.1	id.region	mapped	MAP 1.2.1 (IntersectionGeometry.id.region)
1.2.2	id.id	mapped	MAP 1.2.2 (IntersectionGeometry.id.id)
1.3	revision	mapped	MAP 1.3 (IntersectionGeometry.revision)
1.4.1	status.manualControlsEnabled	static	unset (0)
1.4.2	status.stopTimelsActivated	static	unset (0)
1.4.3	status.failureFlash	converted	Set (1) when: <ul style="list-style-type: none"> WPS.status = "Geel knippen" AND WPS.source = "Procesbesturing" OR "Autonome bewaker"
1.4.4	status.preemptIsActive	static	unset (0)
1.4.5	status.signalPriorityIsActive	static	unset (0)
1.4.6	status.fixedTimeOperation	static	unset (0)
1.4.7	status.trafficDependentOperation	converted	Set (1) when: <ul style="list-style-type: none"> WPS.status = "Regelen"
1.4.8	status.standbyOperation	converted	Set (1) when: <ul style="list-style-type: none"> WPS.status = "Geel knippen" AND NOT WPS.source = "Procesbesturing" OR "Autonome bewaker"
1.4.9	status.failureMode	static	unset (0)
1.4.10	status.off	converted	Set (1) when: <ul style="list-style-type: none"> WPS.status = "Gedoofd"
1.4.11	status.recentMAPmessageUpdate	static	unset (0)

1.4.12	status.recentChangeInMAPAssignedLanesIDsUsed	static	unset (0)
1.4.13	status.noValidMAPisAvailableAtThisTime	static	unset (0)
1.4.14	status.noValidSPATisAvailableAtThisTime	converted	Set (1) when: <ul style="list-style-type: none"> No WPS message has been received yet
1.5	moy	converted	VLOG timestamp (TR + RC delta) as UTC minutes of year
1.6	timeStamp	converted	VLOG timestamp (TR + RC delta) as UTC milliseconds of minute
1.7	enabledLanes	mapped	ITF ControlData Variant.enabledLanes A Variant is deemed active when either: <ul style="list-style-type: none"> The Variant.vlogIndicator matches with the current "VLOG state" One of the Variant.activePeriods is active at the timestamp of the SPAT message Variant activation based on VlogIndicator always supersedes activation based on ActivePeriod
1.8	states	<Level 2>	
1.9	maneuverAssistList	static	<empty>
1.10	regional	static	<empty>
Level 2: MovementState			
2.1	movementName	mapped	ITF ControlData SignalGroup.alias
2.2	signalGroup	mapped	ITF ControlData SignalGroup.signalGroupID
2.3	state-time-speed	<Level 3>	
2.4	maneuverAssistList	static	<empty>
2.5	regional	static	<empty>
Level 3: MovementEvent			

3.1	eventState	converted	In case of no last known/received FT: see "WPS based MovementState" In case of last known/received FT: See "FT based MovementState"
3.2	timing	In case of last known FT: <Level 4> In case of no last known FT: <empty> In case of no "minEndTime": <empty>	
3.3	speeds	static	<empty>
3.4	regional (stateChangeReason)	converted	In case of no last known/received WR: <empty> In case WR message has no bits set: <empty> In case WR message has no mapped "reden wachttijd" set, but one of the unmapped bits is set (bit 12 thru 15): unknown In case of last known/received WR: see "MovementState stateChangeReason".
Level 4: TimeChangeDetails			
4.1	startTime	static	<empty>
4.2	minEndTime	converted	FT.event.minimum Set to <empty> when: <ul style="list-style-type: none"> • FT.event option mask specifies "minimum" as not available • OR FT.event.minimum = -1 • OR it is not possible to represent the time in relation to "moy" <div>  Since this field is mandatory; the entire "TimeChangeDetails" will be set to <empty> in case minEndTime is resolved to <empty> </div>
4.3	maxEndTime	converted	FT.event.maximum Set to <empty> when: <ul style="list-style-type: none"> • FT.event option mask specifies "maximum" as not available • OR FT.event.maximum = -1 • OR it is not possible to represent the time in relation to "moy"

4.4	likelyTime	converted	FT.event.voorspeld Set to <empty> when: <ul style="list-style-type: none"> • FT.event option mask specifies "voorspeld" as not available • OR FT.event.voorspeld = -1 • OR it is not possible to represent the time in relation to "moy"
4.5	confidence	converted	See "FT based MovementState" Set to <empty> when: <ul style="list-style-type: none"> • FT.event option mask specifies "betrouwbaarheid" as not available • OR FT.event.betrouwbaarheid = -1
4.6	nextTime	converted	FT.event.volgende Set to <empty> when: <ul style="list-style-type: none"> • FT.event option mask specifies "volgende" as not available • OR FT.event.volgende = -1 • OR it is not possible to represent the time in relation to "moy"
Level 5: AdvisorySpeed			
Not used			
Level 6: ConnectionManeuverAssist			
Not used			

6.3.6 SPAT payload merging

Since TLC sessions are rate limited, the SPAT payload generation rate should never exceed 10 payloads per second. There is no guarantee that the VLOG device sends a maximum of 10 FT and/or WPS messages per second. To accommodate the possible rate difference, SPAT payloads are merged every 100ms.