**A few quickly created messages for test decoder usage**

A much more complex case developing a MAP message is presented in another document.

**Case #1   
A simple example SPaT Message**

A quick SPaT message. Using the same Dixon Landing intersection and ID (123) and having two active movements. The first being the protected green arrow on the double left hand turn lanes on the west side. The second being the bi-directional Ped crosswalk on the East side. Each “green light/walk” started at 01:05 from the top of the current hour. No “yellow /do not walk” data etc. is provided, nor any future movement/cycle data. [When the example Dixon intersection is completed a full SPaT will be developed to match it to illustrate how signalGroupID is to be used]

The resulting ASN is the file: simpleSPAT.uper

The message is 440 bits (55 bytes) long. The XML is below:

<MessageFrame>

<messageId>19</messageId>

<value>

<SPAT>

<intersections>

<IntersectionState>

<id>

<id>123</id>

</id>

<revision>0</revision>

<status>0000000000000000</status>

<states>

<MovementState>

<movementName>WestSideLeftTurn</movementName>

<signalGroup>1</signalGroup>

<state-time-speed>

<MovementEvent>

<eventState>protected-Movement-Allowed</eventState>

<timing>

<startTime>65</startTime>

<minEndTime>65</minEndTime>

</timing>

</MovementEvent>

</state-time-speed>

</MovementState>

<MovementState>

<movementName>EastSidePedWakl</movementName>

<signalGroup>2</signalGroup>

<state-time-speed>

<MovementEvent>

<eventState>protected-Movement-Allowed</eventState>

<timing>

<startTime>65</startTime>

<minEndTime>85</minEndTime>

</timing>

</MovementEvent>

</state-time-speed>

</MovementState>

</states>

</IntersectionState>

</intersections>

</SPAT>

</value>

</MessageFrame>

**Case #2   
A simple TIM message**

Just some basic road signage in the current TIM message format. Typically this message would be delivered over one of the other channels (not Ch172).

The TIM message allows “frames” of data (TravelerDataFrame ) to be created, here we use just one. We use an ITIS code to encode a speed limit of 25 MPH and center it in a single 50 meter circular region at the intersection at Dixon Landing road developed in other work. The value 32000 defined that the duration is forever (a fixed sign).

A note on converting metric speeds to MPH. Note that the value 782 (15.6464m/s) is defined as equal to 25 mph, see clause 11.5 of the std for normative details and a handy cheat sheet. But in this use case example, we simply use the predefined speed limit phrase sequence defined by J2540 where “25 MPH” is represented by two ITIS codes “n25, mph” or the encoded values of: 12569, 8720

The SSP values that are present in the message have not been used. The details of proper use remain in ballot with the SAE DTSC TC and the J2945/2 draft at this time. Adoption is expected before the end of the year.

The resulting ASN is the file: simpleTIM.uper

The message is 280 bits (35 bytes) long. The XML is below:

<MessageFrame>

<messageId>31</messageId>

<value>

<TravelerInformation>

<msgCnt>0</msgCnt>

<dataFrames>

<TravelerDataFrame>

<sspTimRights>0</sspTimRights>

<frameType>roadSignage</frameType>

<msgId>

<furtherInfoID>0000</furtherInfoID>

</msgId>

<startTime>0</startTime>

<duratonTime>32000</duratonTime>

<priority>0</priority>

<sspLocationRights>0</sspLocationRights>

<regions>

<GeographicalPath>

<description>

<geometry>

<direction>0000000000000000</direction>

<circle>

<center>

<lat>374556439</lat>

<long>-1219169661</long>

</center>

<radius>50</radius>

<units>meter</units>

</circle>

</geometry>

</description>

</GeographicalPath>

</regions>

<sspMsgRights1>0</sspMsgRights1>

<sspMsgRights2>0</sspMsgRights2>

<content>

<speedLimit>

<SEQUENCE>

<item>

<itis>12569</itis>

</item>

</SEQUENCE>

<SEQUENCE>

<item>

<itis>8720</itis>

</item>

</SEQUENCE>

</speedLimit>

</content>

</TravelerDataFrame>

</dataFrames>

</TravelerInformation>

</value>

</MessageFrame>

**Case #3   
A simple RTCM message**

Just some common corrections messages scraped off one of our SNIP NTRIP Casters. Here is a Type 1 message (RTCM 2.x) and a type 1004 message (RTCM 3.x). Only GPS was included (no GLONASS etc.) here. Typically these two “streams” would be delivered as one message over Ch172 at a 1Hz rate. Other supporting RTCM messages (such as orbital data) would be found on other channels. This DSRC message “wraps” the binary message of RTCM differential corrections into a blob. The blob is used by common GNSS devices to achieve higher accuracy.

The resulting ASN is the file: simpleRTCM.uper

The message is 1216 bits (152 bytes) long. The XML is below:

<MessageFrame>

<messageId>28</messageId>

<value>

<RTCMcorrections>

<msgCnt>1</msgCnt>

<rev>unknown</rev>

<msgs>

<RTCMmessage>D300A53EC01125CE2960A0547DB2EA0EB0DFD1EF2040017A3BFDE1E8A178704CD3FE8F7902F413C43FEF0A0BCF1F81A59FF4ABC81E40810FFF78D8003B200385FFA85A40F4034437FB8758BC717FD1CFFD26F206080921BFDE1A56D7CA0328AFE9D71034A0EB35FEE1401A30481008FF49BC813E05337FF7882985F2801FA7FA75840A501E</RTCMmessage>

<RTCMmessage>6641424F4B5848464063</RTCMmessage>

</msgs>

</RTCMcorrections>

</value>

</MessageFrame>

Other message types and representative content can be developed as needed.

**Document History**

Rev 1, just some quick messages to support the Omniair testing request.

Please send comments / suggestions to [davidkelley@itsware.net](mailto:davidkelley@itsware.net)