

# **DATEX II: VMS Publication**

## **Synthesis of the French comments**

### **Introduction**

This paper intends to summarise the different comments received after the VMS Publication proposal has circulated inside the French DATEX Technical Group (GT Datex). It also contains proposals for amending it.

This mirror group is mainly composed of road operators (Ministry and motorway companies) as well as some service providers.

All the materials circulated in DATEX TG have been disseminated in April during three weeks. Four contributions were sent mainly from motorway companies.

### **Utility and global organisation**

Nobody challenges the interest of such publication, which seems to be complementary to the situation publication where a "SignSetting" package already exists.

However, some reactions translate an impression of complexity ("huge labyrinthine system"). Moreover, this publication seems to put together three different datasets with different lifecycles:

- Static (or quasi-static) elements which are frame data like VMS description / location;
- Live information which a wide-area interest like VMS display and/or icons;
- Live information more dedicated to road operators like status.

Therefore it could be wise to split up into two this publication, taking in example what has been done for measured data:

A VMS data publication ("VmsDataPublication") for live information;

A VMS site publication ("VmsSitePublication") for frame data.

Other extensions have been already identified (and perhaps being developed) on similar topics. One can mention for example publications on video cameras or web cams.

Attention must be paid to align the different classes defined in these publications with the situation publication (part "SignSetting") in order to be consistent. However it is also necessary to avoid redundancy and then some constraints/management rules are to add to the model.

As well, the traffic view publication should be modified to be consistent (capability to provide the different VMS messages sequentially on an itinerary).

### **VMS description**

As frame data, this is made through the following classes:

- VMSUnit
- VMS
- Location with ManagedLogicalLocation
- VMSCharacteristics with VMSTextDisplayCharacteristics and VMSPictogramDisplayCharacteristics.

The proposed global organisation is to group these classes into one package corresponding a publication, each VMS unit / VMS being given a reference. This reference would be

afterwards used as an address for exchanging the current VMS text / pictogram display in a more compact and a quicker way.

For mobile VMS, it may be useful to add these characteristics with the text display whereas they units cannot be defined in advance and then referred.

Regarding "VMSUnit", this class combines static information (like "vmsDescription") and dynamic information (like "vmsFault"). Moreover the associated definition to this class is rather ambiguous and can be understood in different ways (VMS controller or physical set (gantry) with several display units).

Regarding "VMS", this class combines static information (like "vmsUnitElectronicAddress") and dynamic information (like "vmsUnitFault").

Regarding "ManagedLogicalLocation" the given definition is unclear and its interest/implementation cases. We are suggesting removing it purely. As well, the different aggregations are cyclic and may lead to inconsistencies between the different occurrences. Last the description provided here may be redundant or inconsistent with the attribute "vmsDescription".

Among the VMS physical types ("vmsPhysicalMounting") the type "access VMS" is missing.

In the class "VMSPictogramDisplayCharacteristics" what is the interest of "pictogramCodeListIdentifier" defined at this level? Generally this pictogram list is defined at the controller level, even if some pictos are not implemented in each VMS. Other possible interpretation: this attribute is a reference to an external standardised picto list: how is it defined and used?

The class "VMSTextDisplayCharacteristics" contains an attribute "maxLuminanceLevel" but not the class "VMSPictogramDisplayCharacteristics": why? How is this luminance level defined? Unit?

## VMS data

See issues on VMSUnit and VMS classes above.

Classes "PictogramDisplayAreaSettings" and "Pictogram" are liased through the same index "PictogramIndex" and one can consider the attributes of the first class could be relevant for the second one. Therefore it is prposed to merge the first class in the second one.

As well, the same merging could be done between "TextDisplayAreaSettings" and "Text" even if only one text is display at the same time.

Class "VMSMessage": several issues are to raise:

- This class contains references to "Situation" and "SituationRecord". One can wonder of such attributes which are useless. Indeed either it used in a specific and isolated publication of VMS messages, and therefore not accessible to the clients who do not have the corresponding situations. Or it used (as planned) in the situation publication (instead of sign settings, and all the relevant information is already coded (redundabcy)).
- This class contain a attribute "associatedManagementOrDiversionPlan" (string type) about the plan identifier associated to the text display. Generally, displaying a text is a measure/action or part of, among several others. Such a plan has no identifier but it is true for measures. One can therefore wonder about the relevance o fit. E.g. displaying a message coming from the "Alp crossing plan" is poor but saying it is based on the measure "XYZ" more relevant. One can imagine some interactions with the work carried out in the TMP activity.
- The attributes "requestedBy" and "setBySystem" are redundant. The first attribute allows precisig "system" or "automatically". Delete the second attribute.

Class "Text": several issues to be raised:

- the proposed model allows for alternating or scrolling messages (each of them being in one language). This model allow displaying the same message in two languages alternatively on the same VMS. But it does not cope with the need to display a short message in two different languages on the same display area without scrolling/alternating effect.
- The attribute "vmsXmlLegend" allows for displaying one message line as ordered. But the multiplicity is [0..1]. Moreover the related definition speaks of "XML tags" for formatting the text. One can wonder it is not rather "HTML tags" that used for such a use.

Class "Pictogram": several issues to be raised:

- When a VMS allows displaying two road signs each beside the other, what is the order of indexing these pictogram. The attribute "relativePositionTo Text" is not enough because the two pictos are on the same side. Moreover, how to do when there is a need to display only pictos, one over each lane (without separate display unit)?
- The attribute "pictogramDescription" is associated with a very rich enum. However it is necessary to assess its consistency to the Vienna convention as well as the national extensions (warning, prohibition and obligation). Re "Special regulation signs" or "Indication signs" (square-shaped) they have to be tailored to actual road layout and may differ from one operator to another. How is this dealt with?

Class "SupplementaryPictogram": several issues to be raised:

- This class should be instantiated to delineate an additional element to the pictogram. This element generally represents with either a text or another pictogram. This class does not seem to allow entering free text (e.g. "Reminder").
- This corresponding enum for the attribute "pictogramDescription" is the same as for the corresponding attribute in the "Pictogram" class. Indeed this picto is much more limited in choice as no road sign representation can be used. In case a second is to display the second area for displaying "Pictogram" is to use.
- According to the Vienna convention on road signs this panel is below the main pictogram. Thus the attribute "relativePositionToMainPictogram" seems useless.

## Rules consistency

The different class names beginning with VMS do not comply the upper case notation.

The classes "TextDisplayAreaSettings" and "PictogramDisplayAreaSettings" have the same attribute names but definition different. It is not consistent with the global scope of attribute definition of DATEX II.

As well "VMSTextDisplayCharacteristics" and "VMS PictogramDisplayCharacteristics" share some common attributes but with different definition.

As well "Pictogram" and "SupplementaryPictogram" share some common attributes but with different definition.

Why are the names of the classes "TextDisplayAreaSettings" and "PictogramDisplayAreaSettings" not prefixed with "VMS"? Idem for "Text" and "Pictogram" of the package "VMSMessage" that are not reusable classes?

Why does the attribute name "vmsPhysicalMounting" (in class "VMSCharacteristics") not fit the corresponding enum name ("PhysicalMountingEnum")?

Why does the attribute name "textInformationType" (in class "Text") not fit the corresponding enum name ("VMSTextInformationTypeEnum")?