



Annexe 10 : Construction des traces et de l'history des événements

Livrable de la famille 2.4.1.4_M

Activité 2: Etudes

Sous-activité 2.4 > Specifications

Version 0.10

Date Publication: 08/09/2022



Co-financed by the Connecting Europe
Facility of the European Union

The contents of this publication are the sole responsibility of the French C-ITS project partners and do not necessarily reflect the opinion of the European Union.

Document: 2.4.1.4_M Annexe 10

Responsable, entité : Vincent Robin, CEREMA

Status: Version 0.10, approved

Distribution

Date	Version	Auteur	MAJ	Diffusion
11/04/2022	0.01	V.Robin	Initialisation du document sur la base du document « SCOOP _ Description des traces et History » rédigé par E.Petit	
08/09/2022	0.10	V.Robin	Prise en compte des remarques de relecture suite au COCSIC de mai 2022	COCSIC

Table des matières

TABLE DES MATIERES	3
1 INTRODUCTION	4
1.1 PRECAUTION DE LECTURE	4
2 MESSAGE DESCENDANT (SAGT=> UBR=> UEV)	5
3 MESSAGES MONTANTS (UEV=>SAGT)	9
3.1 CAS D'USAGES CONCERNES	9
3.2 TRACE ET HISTORY SE SUPERPOSENT, L'EVENTPOSITION SE DEPLACE.	9
3.3 TRACE ET HISTORY NE SE SUPERPOSENT PAS, L'EVENTPOSITION EST FIXE.	11
3.4 TRACE EXISTE, EVENTHISTORY N'EST PAS PRESENT, L'EVENTPOSITION EST FIXE.	12

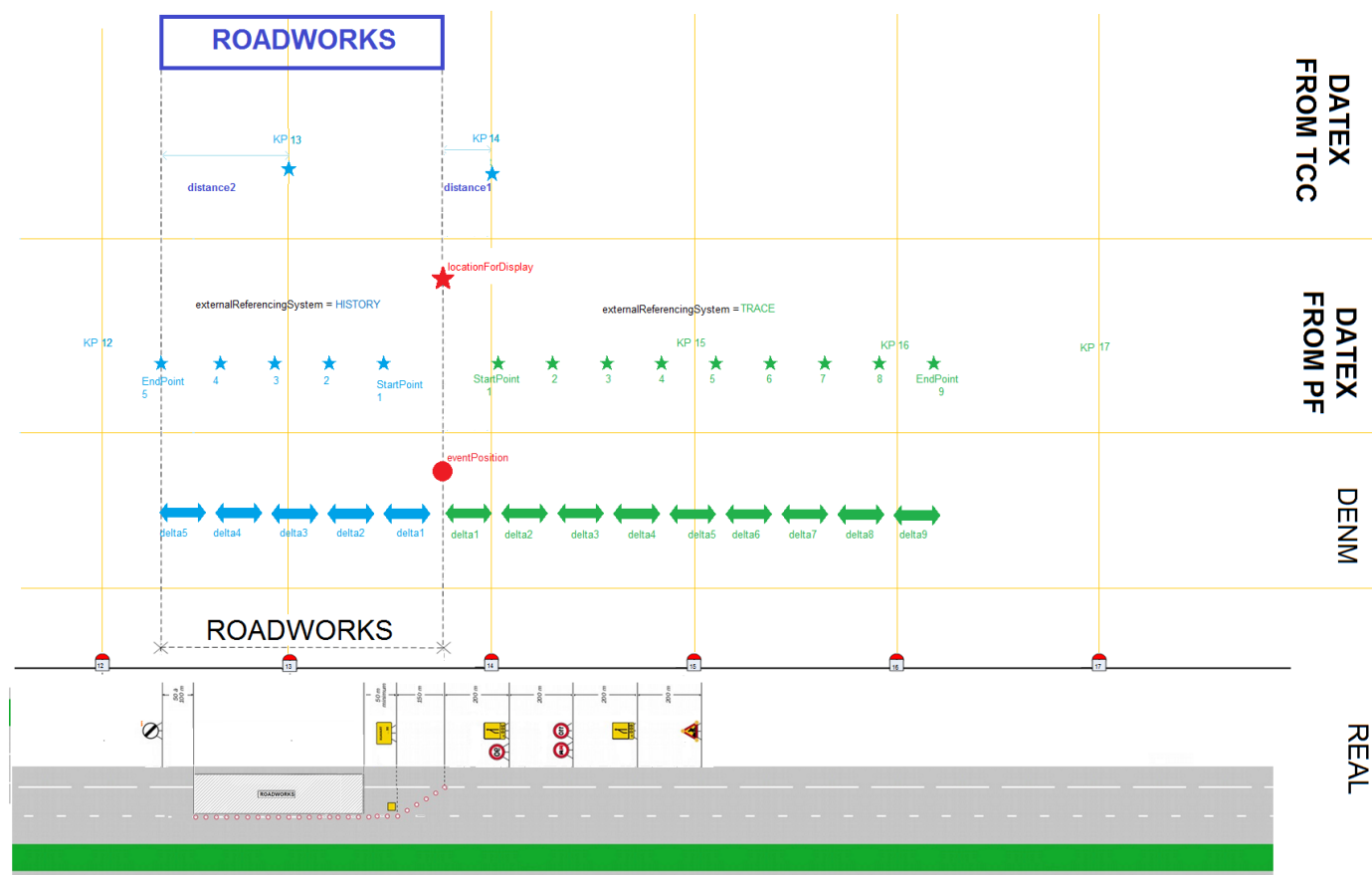
1 Introduction

1.1 Précaution de lecture

Les simplifications suivantes sont prises dans le document :

- Les points s'appliquant aux « UBR » s'appliquent aussi au nœud national, et aux UEVG en mode « UBR mobile ».
- On ne parle pas d'altitude, ni d'informationquality, ni du Δ pathtime
- lat = latitude, lata= latitude du point a, lat1 = latitude du point 1. Y est la latitude du point (X,Y)
- lon = longitude, lona= longitude du point a, lon1 = longitude du point 1. X est la longitude du point (X,Y)

2 Message descendant (SAGT=> UBR=> UEV)



2.1.1 Etape 1 : génération par le SAGT

Le SAGT émet un message contenant le linéaire de l'événement.

Le « groupOfLocation » est un linéaire (par exemple « linearWithinLinearElement ») tel que :

- Attributs :
 - o directionRelativeOnLinearSection est précisé :
 - opposite = sens inverse des PR croissants
- Précisant le linearElement concerné (« A8 » par exemple)
- fromPoint de type DistanceFromLinearElementReferent avec :
 - o distanceAlong = distance1 (dans le sens indiqué en directionRelativeOnLinearSection)
 - o fromReferent referentIdentifier = KP14
- toPoint de type DistanceFromLinearElementReferent avec :
 - o distanceAlong = distance2 (dans le sens indiqué en directionRelativeOnLinearSection)
 - o fromReferent referentIdentifier = KP13

2.1.2 Etape 2 : complément par la plateforme

• **Précaution**

Une évolution pour la création des traces et eventhistory sera faite pour améliorer la qualité en fonction du linéaire. Les interdistances devront être réduites dans les courbes et devront être augmentées dans les lignes droites.

• **pour l'eventHistory**

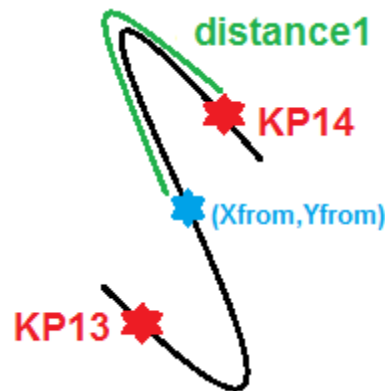
La plateforme recalcule les coordonnées en (x,y) du linéaire de l'évènement.

Le premier point du linéaire de l'évènement est (Xfrom,Yfrom) avec :

- Xfrom = la longitude du point situé sur la chaussée à une distance de distance1 du point KP14, dans le sens des PR décroissants
- Yfrom = la latitude du point situé sur la chaussée à une distance de distance1 du point KP14, dans le sens des PR décroissants

Le dernier point du linéaire de l'évènement est (Xto, Yto) avec :

- Xto = la longitude du point situé sur la chaussée à une distance de delta2 du point KP13, dans le sens des PR décroissants
- Yto = la latitude du point situé sur la chaussée à une distance de delta2 du point KP13, dans le sens des PR décroissants



1 : La distance est curviligne et non à vol d'oiseau.

La plateforme calcule ensuite des points à intervalles réguliers entre ces deux points : tous les intervalles pour l'History sont les mêmes.

La plateforme crée alors le linéaire HISTORY dans le DATEX comme suit :

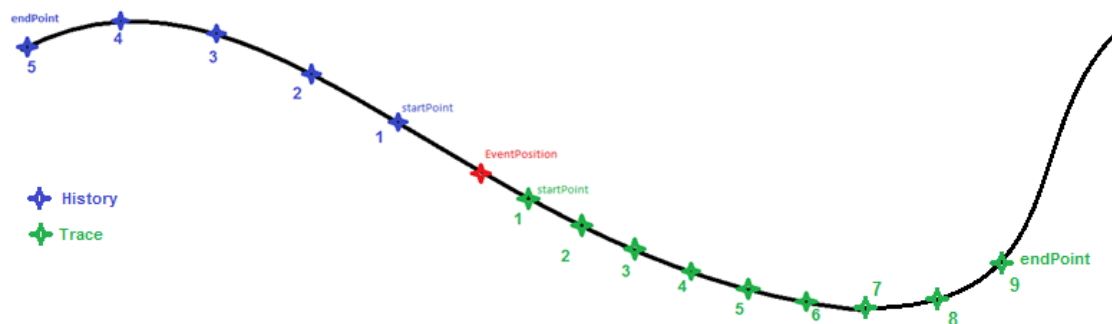
- Location for display = le point (Xfrom, Yfrom)
- Le startPoint du linéaire « HISTORY » est le premier point calculé après le (Xfrom,Yfrom), soit le plus proche de l'eventposition
- Le endPoint = le point (Xto, Yto) (le plus loin de l'eventposition)
- Les intermediatePoint sont les points intermédiaires, depuis le startPoint jusqu'au endPoint

• **pour la trace**

Une fois que la plateforme a trouvé la position du premier point de l'évènement : (Xfrom,Yfrom), elle calcule tous les chemins qui mènent à l'évènement, et d'une longueur maximale fixée par un paramètre, et les découpe selon des points régulièrement positionnés à un intervalle identique.

Elle crée alors le (ou les) linéaires « TRACE » comme suit :

- Le startPoint de la TRACE est le premier point situé à l'intervalle calculé de cet ensemble, le plus proche de l'eventposition
- Les intermediatePoint sont les points intermédiaires depuis le startPoint jusqu'au endPoint
- Le endPoint est le dernier point de cette série de point, le plus loin de l'eventposition



2 Illustration du positionnement des points dans un message DATEX

• Message résultant :

La plateforme modifie le groupOfLocation pour avoir un groupOfLocation, de type « NonOrderedLocationGroupByList », contenant plusieurs linéaires (au moins 2) :

- Le premier « linearWithinLinearElement »
 - Attributs :
 - externalReferencingSystem= HISTORY
 - index =1
 - "PointCoordinates" with the relation "locationForDisplay" :
 - Lon = Xfrom
 - Lat = Yfrom
 - startPointOfLinearElement :
 - referentIdentifier = 1
 - Lon(1BLEU)
 - Lat(1BLEU)
 - intermediatePointOfLinearElement)
 - referentIdentifier = 2
 - Lon(2BLEU)
 - Lat(2BLEU)
 - intermediatePointOfLinearElement)
 - referentIdentifier = 3
 - Lon(3BLEU)
 - Lat(3BLEU)
 - intermediatePointOfLinearElement)
 - referentIdentifier = 4
 - Lon(4BLEU)
 - Lat(4BLEU)
 - endPointOfLinearElement
 - referentIdentifier = 5
 - Lon(5BLEU) = Xto
 - Latitude(5BLEU) = Yto
- Le second « linearWithinLinearElement »
 - Attributs :
 - externalReferencingSystem= TRACE
 - index = 2
 - startPointOfLinearElement :
 - referentIdentifier = 1
 - Lon(1VERT) = Xfrom
 - Lat(1VERT) =Yfrom
 - intermediatePointOfLinearElement
 - referentIdentifier = 2
 - Lon(2VERT)
 - Lat(2VERT)
 - intermediatePointOfLinearElement
 - referentIdentifier = 3
 - Lon(3VERT)
 - Lat(3VERT)

- intermediatePointOfLinearElement
 - referentIdentifiant = 4
 - Lon(4VERT)
 - Lat(4VERT)
- Etc.
- endPointOfLinearElement
 - referentIdentifiant = 9
 - Lon(9VERT)
 - Lat(9VERT)
- Eventuellement d'autres linéaires

2.1.3 Etape 3 : traduction par l'UBR

A la réception du message de la plateforme, l'UBR crée alors un message DENM avec les champs suivants :

- EventPosition = (Xfrom,Yfrom)
- History :
 - (lat(1BLEU)-Xfrom, lon(1BLEU)-Yfrom)
 - (lat(2BLEU)- lat(1BLEU), lon(2BLEU)-lon(1BLEU))
 - (lat(3BLEU)- lat(2BLEU), lon(3BLEU)-lon(2BLEU))
 - (lat(4BLEU)- lat(3BLEU), lon(4BLEU)-lon(3BLEU))
 - (lat(5BLEU)- lat(4BLEU), lon(5BLEU)-lon(4BLEU))
- Trace :
 - (lat(1VERT)-Xfrom, lon(1VERT)-Yfrom)
 - (lat(2VERT)- lat(1VERT), lon(2VERT)-lon(1VERT))
 - (lat(3VERT)- lat(2VERT), lon(3VERT)-lon(2VERT))
 - (lat(4VERT)- lat(3VERT), lon(4VERT)-lon(3VERT))
 - ...
 - (lat(9VERT)- lat(8VERT), lon(9VERT)-lon(8VERT))
- Eventuellement d'autres traces

Note : les frompoint et topoint éventuellement présents dans les linéaires venant de la PF, ne sont pas utilisés par l'UBR.

3 Messages montants (UEV=>SAGT)

3.1 Cas d'usages concernés

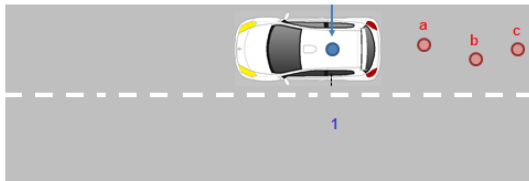
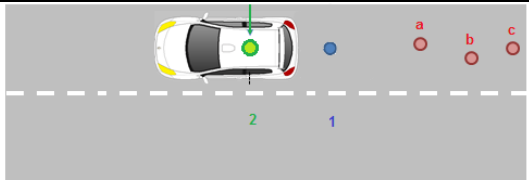
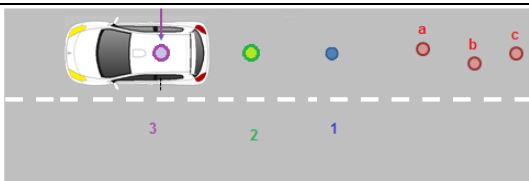
Les cas d'usage concernés correspondent aux critères ci-dessous :

- Sans eventhistory
 - Des cas d'usage travaux
 - Les événements mobiles avec 1 trace et 1 eventposition mis à jour en fonction du déplacement du véhicule.
 - Les événements ponctuels et manuels : 1 eventposition fixe et 1 trace
- Avec eventhistory
 - 1 trace, 1 eventhistory qui se superposent

3.2 Trace et History se superposent, l'eventPosition se déplace.

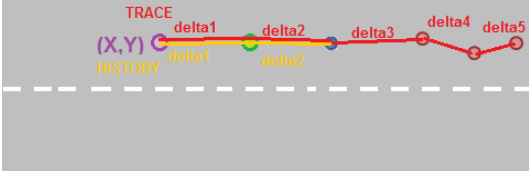
3.2.1 Etape 1 : génération par l'UEV

The event position changes at each update and the previous event position become point of the eventhistory. Trace and eventhistory overlap.

Schéma	Eventposition	Trace	History
	Le point bleu : (lat1,lon1)	[(lata-lat1,lonal-lon1), (latb-lata,lonb-lona), (latc-latb,lonc-lonb)]	Aucun
	Le point vert : (lat2,lon2)	[(lat2-lat1,lon2-lon1), (lata-lat1,lonal-lon1), (latb-lata,lonb-lona), (latc-latb,lonc-lonb)]	[(lat2-lat1,lon2-lon1)]
	Le point violet : (lat3,lon3)	[(lat3-lat2,lon3-lon2), (lat2-lat1,lon2-lon1), (lata-lat1,lonal-lon1), (latb-lata,lonb-lona), (latc-latb,lonc-lonb)]	[(lat2-lat1,lon2-lon1), (lat3-lat2,lon3-lon2)]

3.2.2 Etape 2 : Traduction par l'UBR :

L'UBR reçoit ce type de DENM :

Schéma	EventPosition	Trace	History
	Le point violet : (X,Y)	[(delta1rougex, delta1rougey), (delta2rougex, delta2rougey), (delta3rougex, delta3rougey), (delta4rougex, delta4rougey), (delta5rougex, delta5rougey)]	[(delta1jaunex, delta1jauney), (delta2jaunex, delta2jauney)]

		delta3rougey) , (delta4rougey, delta4rougey) , (delta5rougey, delta5rougey)]	
--	--	--	--

L'UBR crée alors un message DATEX, contenant un groupOfLocation, de type « NonOrderedLocationGroupByList », contenant 2 linéaires :

- Le premier « linearWithinLinearElement »
 - o Attributs :
 - externalReferencingSystem= HISTORY
 - index =1
 - o "PointCoordinates" with the relation "locationForDisplay" :
 - Lon = X
 - Lat = Y
 - o startPointOfLinearElement :
 - referentIdentifiant = 1
 - Lon(1H) =x+ delta1jaunex
 - Lat(1H) = y+ delta1jaune
 - o (et dans des cas plus complexes : n intermediatePointOfLinearElement)
 - o endPointOfLinearElement
 - referentIdentifiant = 2
 - Lon(2H) = Longitude(1H) + delta2jaunex
 - Latitude(2H) = Latitude(1H) + delta2jaune
- Le second « linearWithinLinearElement »
 - o Attributs :
 - externalReferencingSystem= TRACE
 - index = 2
 - o startPointOfLinearElement :
 - referentIdentifiant = 1
 - Lon(1T) =x+ delta1rouge
 - Lat(1T) = y+ delta1rouge
 - o intermediatePointOfLinearElement
 - referentIdentifiant = 2
 - Lon(2T) = Lon(1T) +delta2rouge
 - Lat(2T) = Lat(1T) +delta2rouge
 - o intermediatePointOfLinearElement
 - referentIdentifiant = 3
 - Lon(3T) = Lon(2T) +delta3rouge
 - Lat(3T) = Lat(2T) +delta3rouge
 - o intermediatePointOfLinearElement
 - referentIdentifiant = 4
 - Lon(4T) = Lon(3T) +delta4rouge
 - Lat(4T) = Lat(3T) +delta4rouge
 - o endPointOfLinearElement
 - referentIdentifiant = 5
 - Lon(5T) = Lon(4T) +delta5rouge
 - Lat(5T) = Lat(4T) +delta5rouge

3.2.3 Etape 2 : Traduction par PFro :

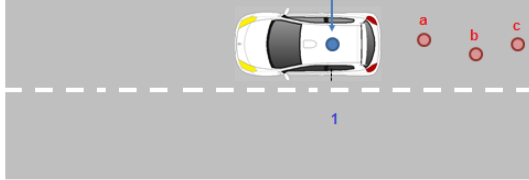
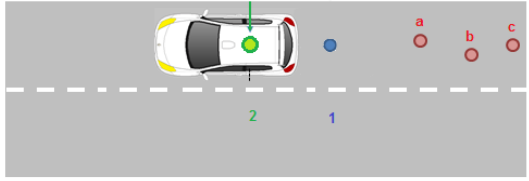
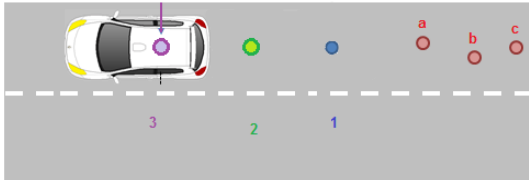
PFro n'exploite pas les traces et transforme l'history en PR ou TPEG pour le SAGT.

3.3 Trace et History ne se superposent pas, l'eventPosition est fixe.

3.3.1 Etape 1 : génération par l'UEV

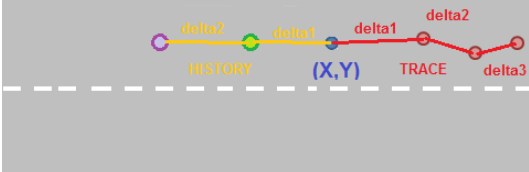
The event position remains identical at each update and new event history points are added along with each update.

Ce qui est illustré par le tableau suivant :

Schéma	Eventposition	Trace	History
	Le point bleu : (lat1,lon1)	[(lata-lat1,lona-lon1), (latb-lata,lonb-lona), (latc-latb,lonc-lonb)]	Aucun
	Le point bleu : (lat1,lon1)	[(lata-lat1,lona-lata), (latb-lata,lonb-lona), (latc-latb,lonc-lonb)]	[(lat2-lat1,lon2-lat1)]
	Le point bleu : (lat1,lon1)	[(lata-lat1,lona-lata), (latb-lata,lonb-lona), (latc-latb,lonc-lonb)]	[(lat2-lat1,lon2-lat1),(lat3-lat2,lon3-lat2)]

3.3.2 Etape 2 : Traduction par l'UBR :

L'UBR reçoit un DENM de ce type :

Schéma	EventPosition	Trace	History
	Le point bleu : (X,Y)	[(delta1rouge), (delta2rouge), (delta3rouge)]	[(delta1jaune), (delta2jaune)]

L'UBR crée alors un message DATEX, contenant un groupOfLocation, de type « NonOrderedLocationGroupByList », contenant 2 linéaires :

- Le premier « linearWithinLinearElement »
 - o Attributs :
 - externalReferencingSystem= HISTORY
 - index =1
 - o "PointCoordinates" with the relation "locationForDisplay" :
 - Lon = X
 - Lat = Y
 - o startPointOfLinearElement :
 - referentIdentifier = 1

- $Lon(1H) = x + \delta 1_{jaunex}$
 - $Lat(1H) = y + \delta 1_{jauney}$
- (et dans des cas plus complexes : n intermediatePointOfLinearElement)
- endPointOfLinearElement
 - referentIdentifiant = 2
 - $Lon(2H) = Longitude(1H) + \delta 2_{jaunex}$
 - $Latitude(2H) = Latitude(1H) + \delta 2_{jauney}$
- Le second « linearWithinLinearElement »
 - Attributs :
 - externalReferencingSystem= TRACE
 - index = 2
 - startPointOfLinearElement :
 - referentIdentifiant = 1
 - $Lon(1T) = x + \delta 1_{rougex}$
 - $Lat(1T) = y + \delta 1_{rougey}$
 - intermediatePointOfLinearElement
 - referentIdentifiant = 2
 - $Lon(2T) = Lon(1T) + \delta 2_{rougex}$
 - $Lat(2T) = Lat(1T) + \delta 2_{rougey}$
 - endPointOfLinearElement
 - referentIdentifiant = 3
 - $Lon(3T) = Lon(2T) + \delta 3_{rougex}$
 - $Lat(3T) = Lat(2T) + \delta 3_{rougey}$

3.3.3 Etape 2 : Traduction par PFro :

PFro n'exploite pas les traces et transforme l'history en PR ou TPEG pour le SAGT.

3.4 Trace existe, EventHistory n'est pas présent, l'eventPosition est fixe.

Ce CU est possible, par exemple dans le cas d'un accident. On est alors dans le cas d'un événement ponctuel sans History.