Working Group 3

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Author : | | Christian GIRAUD | | Date : | 04/07/2014 | | | | | |
|  | | | | | | | | | | |
| **Subject :** | | | | | | | Document Review : | | 🗹 | |
| **OpenETCS WP3 Task Force** | | | | | | | Design Review : | | ❒ | |
| WG 3 Architecture - provisional - | | | | | | | Other : | |  | |
|  | |  | | | | | | | | |
|  |  | |  | | |  | |  | |  |
|  |  | |  | | |  | |  | |  |

|  |  |  |
| --- | --- | --- |
| **Name** | **Position** | **Company / Department** |
| Valerio Raimondi |  | Alstom |
| Stephane Besure |  | Alstom |
| Nicolas Boverie |  | Alstom |
| Christian Giraud |  | Alstom |
| Yoann Guyot |  | Cetic |
| Baseliyos Jacob |  | DB |
| Bernd Hekele |  | DB |
| Niklas Schaffrath |  | Siemens |
| Uwe Steinke |  | Siemens |
| Jos Holtzer |  | NS |
| Jan Welvaarts |  | L’loyds Register Rail |
| Sylvain Baro |  | SNCF |
| Marielle Petit-Doche |  | Systerel |
| Jan Welte |  | TU Braunschweig |
|  |  |  |
|  |  |  |

**Distribution to:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Position** | **Company / Department** |
|  |  | Alstom TIS |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Table of Content

1. Objectives and strategy 5

2. Context Of Open-ETCS 6

2.1 references 6

2.2 High Level Architecture 6

2.3 Concept Of Open ETCS 7

2.3.1 Unisig Specification 8

2.3.2 Signalling Application Specification 8

2.3.3 Formal Specification 8

2.3.4 Model 8

2.3.5 Kernel Code 8

2.3.6 Virtual Machine 8

2.3.7 Host Machine 8

2.3.8 Real Time Machine 8

2.3.9 Validation 8

3. Data for Safe Movement 9

3.1 General 9

3.2 Linking and Re-Positioning 9

3.3 Basic Track Description 11

3.3.1 Movement Authority 11

3.3.2 Standard Speed Profile 11

3.3.3 Gradients Profile 12

3.4 Optional Track Description 12

3.5 Other Track to Train Data 14

3.6 Radio Communication with RBC. 14

3.6.1 Train To Track Packets for Radio 15

3.6.2 Track to Train Message 15

3.6.3 Train To Track Radio Message 16

3.7 System Data 16

3.7.1 Fixed Values 16

3.7.2 National Values 17

3.7.3 Train Data 17

3.7.4 Additional Data 17

3.7.5 Date and Time 17

3.7.5.1 UTC 17

3.7.6 Data view 18

4. Processes Over Data 20

4.1 Data Preliminary Operation 20

4.2 Storage of Data 23

4.3 Update of Data 25

4.4 Train Positioning and linking 25

4.5 Emergency Stop 29

4.6 Speed and Distance Monitoring 29

4.6.1 Computation of Deceleration and Brake Build up Time 29

4.6.2 MRSP computation 29

4.6.3 Computation of Target and Curves 29

4.6.4 Supervision Limits Computation 29

4.6.4.1 Ceiling Speed Monitoring 29

4.6.4.2 Target Speed Monitoring 29

4.6.4.3 Release Speed Monitoring 29

4.7 Protection against undesirable movement 29

4.7.1 Roll Away 29

4.7.2 Reverse 29

4.7.3 Standstill 29

4.8 Mode Management 29

4.9 Brake command handling 29

4.10 Special Function 29

4.11 Version Management 29

5. SysML First Draft 30

5.1 First Level IBD 30

5.2 Second Level 31

5.3 To Position Train 34

5.4 To Achieve Processes 36

5.5 Breakdown Structure of EVC 37

5.5.1 General 37

5.5.2 First Level 37

5.5.3 Second Level 37

6. Development of A5.1 ( example) 39

6.1 references 39

6.2 overview 39

6.3 Context of Speed and Distance Monitoring 39

6.4 Example of Balise Group in level 1 41

6.5 EB Supervision Curve 41

6.6 EB Supervision Computation 42

6.7 EB Supervision Excel File 43

6.8 EB Supervision per Quantum 46

6.9 Additional Issue on EB Supervision per Singular Point 49

# Objectives and strategy

The main objective is to propose a technical plan for Open-ETCS in relation with UNISIG specification.

The aim of Open-ETCS is to insure that “Subset\_026 version\_3.3.0” shall be implemented and interpreted following the same understanding by every supplier of ETCS devices.

Given the difficulties, it has been defined that consists in working on application software of EVC. Trackside application software is anticipated as enough transparent from EVC.

Three phases are involved :

* **First Phase** : The Subset-026 chapter3 being basically a system requirement definition, it needs to be analysed and structured through SysML methodology or equivalent. That permits to build a breakdown structure as follows :
  + Basic Software and hardware, is proprietary software of each ETCS supplier,
  + Application Software (so-called “kernel software”),
    - Balise Data storage and update,
    - Positioning,
    - Speed Control Monitoring,
    - etc..
  + Application Program Interface : Alstom solution is acting as reference.
* **Second Phase** : The Breakdown Structure of Application Software is refined up to reach a level of detail that permits to proceed to a formal (non ambiguous) specification and encoding.
  + This phase leads to include most of the modules within the Open-ETCS library.
  + On a first step, it has been agreed to work on “Balise Data Storage” and “Positionning”.
  + It may appear the need to define a database structure when study of other functions.
  + At the end of this phase, “Kernel Software” is ready for integration on “Host Machine” or on “Real Time Machine”.
* **Third Phase** : In order to consolidate the specification, a model is developed for the most complex functions that permits to get a reference for each signalling implementation. This is the concept of Virtual Machine.

# Context Of Open-ETCS

## references

UNISIG Subset\_026 version\_3.3.0

Chapter 3 : ERTMS / ETCS Principes

Chapter 5 : ERTMS / ETCS Procedures

Chapter 7 : ERTMS / ETCS Language

## High Level Architecture

The following Block Diagram Definition (BDD) gives an abstract of global ETCS from the on-board (EVC) point of view :

Balise

Odometry

DMI

TIU

CORE

Radio

(RBC)

EVC

**Block Diagram Definition**

EVC is composed of :

* Balise reception
* Odometry
* DMI : Display Man Interface
* TIU : Train Interface Unit
* CORE : Computer
* Radio sub-system (GSM) and RBC (Radio Block Center)
* Euroloop (optional)
* Recorder Unit

Communication are insured through a triple Bus (CAN bus for instance).

Specification of communication are defined by API (Application Program Interface) from functional point of view.

## Concept Of Open ETCS

Open ETCS aims to provide a link between the UNISIG Specification (Mainly Subset\_026) and any implementation of On-board provider.

For reaching this challenge, a double activity diagram is proposed as follows :

Unisig Specification V3.3.0

Formal Specification

(SysML & Software Spec)

Model

Kernel Code

Host Machine

(Soft ADA C++)

Real Time Machine

(hard +soft + simu)

Virtual Machine

( Excel )

Signalling Implementation Specification

Validation

Hardware + basic soft + simulator

**Block “Actigram”**

This actigram shows the 3 branches of activity that are involved in order to validate any Signaling Application.

### Unisig Specification

The main document of reference is the subset 026 in version V3.3.0.

All chapters are applicable and all subset that are referenced in it.

### Signalling Application Specification

The Signalling Application Specification should include :

* track layout with balise group (BG) and signals,
* definition of each BG coverage, in normal and reverse direction if needing,
* definition of mandatory data for each BG,
* definition of optional data for each BG,
* definition of system data.

### Formal Specification

The Formal Specification shall include two steps :

* SysML defines the architecture & Software Spec,
* Formal language specifies the software or can be used as retro engineering.

### Model

The use of “Excel” is a solution to study braking curves of each signal and for a run of a virtual train.

Other solution ?

### Kernel Code

Programing language : ADA vs C++.

Formal language : B language.

Bridge to pass from B to ADA.

### Virtual Machine

Excel, solution linked to Model.

### Host Machine

Solution to compare result with virtual machine.

### Real Time Machine

Idem Host.

### Validation

Automatic validation is interesting for checking in case of software updating.

# Data for Safe Movement

## General

This chapter aims to provide an overview of EVC Data Packets and Variables following the definition hereafter :

* Data Packets related to Linking,
* Data Packets related to basic track description,
* Data Packets related to optional track description,
* Data Packets and Variables related to radio communication with RBC,
  + Additional Data Packets for radio communication
  + Track To Train Radio Data Messages,
  + Train To Track Radio Data Messages.

All Data are structure of variables as defined hereafter (see Subset-026, chapter 8) :

* A **Telegram** is a set of Packet transmitted through one balise which includes a “header”,
* A **Balise Message** is a set of all Telegrams of a BG,
* A **Radio Message** is a set of Variables and Packets transmitted through GSM\_R and includes a “header”,
* A **Packet** is a set of variables which may include option and repeating.

## “Linking” versus “Linked” and Re-Positioning

### Definition

“Linking” data are playing a major role in management of track description, that is :

* To know in advance which BG will be met from any current BG (several BG can be defined),
* To check the train position regarding the window related to the last BG position,
* To check the train position regarding the route provided by the previous BG,
* To up-date the track description provided by the previous BG,
* To reduce the train position error to the position inaccuracy of the last BG,
* To permit or not an immediate reaction in case of BG missing.

But the use of linking is only an option of design, and depending on the confidence in odometry. As result, “Linking” is submitted to a qualifier meaning “Linking Use” or “Linking Not Used”. All information about “Linking” are given and described within SRS chapter 3-4-4.

“Linked” is a qualifier included within the header of any balise group and which defines the nature of data that are included in it :

* “Linked” means that BG :
  + can be announced in the current LRBG to up-date data.
  + can become the new LRBG if consistent with opened window,
  + can be not announced but may still become the new LRBG without checking.
  + if not announced, position is given by train position and current LRBG.
* “Unlinked” means that BG :
  + can be announced in the current LRBG, to provide additional data.
  + cannot become the new LRBG even if consistent with opened window,
  + can be not announced but may still provide additional data.
  + if not announced, position is given by train position and current LRBG.

Re-Positioning is involved to reduce positioning error when needing.

Linking is defined by packet 5 for all level

Linked / Unlinked is defined by one header qualifier.

Repositioning is defined by packet 16 :

* only through balise in 1 level.

### True Table

When “Linking” is used, the true table is :

|  |  |  |
| --- | --- | --- |
| **LINKING USE on-board** | **Linked BG** | **Unlinked BG** |
| **Linking exists in message** | Window is computed  Train Position is checked  BG becomes LRBG if checking OK  & Downstream Data are up-dated | Window is computed  Train Position is checked  Additionnal Data are stored if checking OK |
| **No Linking in message** | No action | Additionnal Data are stored |

When “Linking” is not used, the true table is :

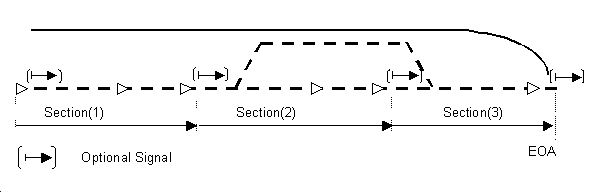
|  |  |  |
| --- | --- | --- |
| **LINKING NOT USE** | **Linked BG** | **Unlinked BG** |
| **Linking exists** | Train Position gives BG position  BG becomes LRBG  & Downstream Data are up-dated | Train Position gives BG position  Additionnal Data are stored |
| **No Linking** | Train Position gives BG position  BG becomes LRBG  & Downstream Data are up-dated | Train Position gives BG position  Additionnal Data are stored |

## Basic Track Description

“Movement Authority”, “Standard Speed Profile” and “Gradient Profile” are the basic element for a Track Description. These data are provided by “Linked BG” or by Radio.

### Movement Authority

* Section : the MA is composed of several section with optional timer to enter,
* LOA : is the length of MA starting from LRBG, and is terminated by EOA,
  + must be covered by SSP and Gradient Profile,
* DP : Danger Point is beyond the EOA and permits to approach EOA,
* OL : Overlap is beyond the EOA and requires a locking from some other routes.
* Veoa : is the limit speed related to EOA only with balise MA level 1.

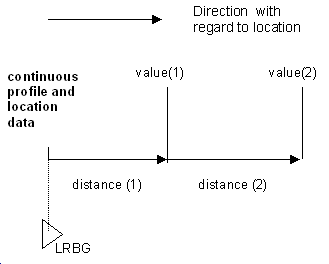


MA level 1 is defined by packet 12 through balise (includes Veoa).

MA level 2/3 is defined by packet 15 through radio (no Veoa).

### Standard Speed Profile

SSP is a continuous profile defined by a set of : {distance, value}.



SSP is defined by packet 27 through balise in level 1.

SSP is defined by packet 27 through radio in level 2/3.

### Gradients Profile

Gradient Profile is a continuous profile defined by a set of {distance, value}.

Gradient Profile describes the worst case of grade over the maximum length of train, as follows :

* Unit is 1/1000,
* Value positive for up-hill,
* Value negative for down-hill.

Gradient Profile is defined by packet 21 through balise in level 1.

Gradient Profile is defined by packet 21 through radio in level 2/3.

## Optional Track Description

The following packets are related to control speed (See Subset-026, Chapter 3.11).

* Axle Speed Profile is defined through packet 61.
* Level Transition Order is defined through packet 41.
* Conditional Level Transition Order is defined through packet 46.
* Temporary Speed Restriction is defined through packet 65.
* Mode Profile and Related Speed Restriction is defined through packet 80.
* LX Speed Restriction is defined through packet 88.
* Speed Restriction to ensure Permitted Braking Distance is defined through packet 52.

The following packets are related to DMI (See Subset-026, Chapter 3.12).

* Track Conditions are defined through packet 39, 40, 67, 68, 69.
* Route Suitability is defined through packet 70.
* Text Transmission is defined through packet 72 & 76.
* Geographical Position is defined through packet 79.

## Other Track to Train Data

Most of the following packets are inserted in radio messages and used for miscellaneous functions (See subset-026, chapter 4.8.3).

* SR Distance from loop ( packet 13 ).
* Session Management ( packet 42 ).
* Radio Network Registration ( packet 45 ).
* MA Request Parameters ( packet 57 ).
* Position Report Parameters ( packet 58 ).
* Inhibition of revocable TSR ( packet 64 ).
* RBC Transition Order( packet 131 ).
* Danger for SH ( packet 132 ).
* Radio Infill Area Info ( packet 133 ).
* End Of Loop Management ( packet 134 ).
* Stop Shunting ( packet 135 ).
* Infill Location Reference ( packet 136 ).
* Stop if in SR Mode ( packet 137 ).
* Reversing Area Information ( packet 138 ).
* Reversing Supervision Information ( packet 139 ).
* Train Running Number from RBC ( packet 140 ).
* Default Gradient for TSR ( packet 141 ).
* Session Management with RIU ( packet 143 ).
* Inhibition of BG consistency reaction ( packet 145 ).
* Default Balise / Loop / RIU ( packet 254 ).
* End of Telegram / Message ( packet 255 ).

## Radio Communication with RBC.

The Radio Communication messages are described in 2 tables hereafter (See subset-026, chap 4.8.3 & chap 3.8.3).

The Radio Communication with the RBC involves Radio Messages in both direction (Train To Track and Track To Train) and uses the same variables as those of Balise Communication.

Then, a list of new Data Packets is defined in order to address the “Track To Train” radio message.

### Train To Track Packets for Radio

* Position Report ( packet 0 ).
* Position Report based on two balise groups ( packet 1 ).
* Onboard telephone numbers ( packet 3 ).
* Error Reporting ( packet 4 ).
* Train running number from EVC ( packet 5 ).
* Level 2/3 transition information ( packet 9 ).
* Validated train data ( packet 11 ).
* Data used by applications outside the ERTMS/ETCS system ( packet 44 ).
* End of Telegram / Message ( packet 255 ).

### Track to Train Message

| **Mess. Id.** | **Message Name** | **From** | **Packets / Variables**  ( OP = optional packets) |
| --- | --- | --- | --- |
| 2 | SR Authorisation + optional list | RBC | T\_TRAIN, M\_ACK, NID\_LRBG + OP + D\_SR |
| 3 | Movement Authority | RBC | T\_TRAIN, M\_ACK, NID\_LRBG + OP + 15 |
| 6 | Recognition of exit from TRIP mode | RBC | T\_TRAIN, M\_ACK, NID\_LRBG |
| 8 | Acknowledgement of Train Data | RBC | T\_TRAIN, M\_ACK, NID\_LRBG + T\_TRAIN of train |
| 9 | Request to Shorten MA | RBC | T\_TRAIN, M\_ACK, NID\_LRBG + OP + 15 + 80 |
| 15 | Conditional Emergency Stop | RBC | T\_TRAIN, M\_ACK, NID\_LRBG + D\_EMERGENCY |
| 16 | Unconditional Emergency Stop | RBC | T\_TRAIN, M\_ACK, NID\_LRBG + NID\_EM |
| 18 | Revocation of Emergency Stop | RBC | T\_TRAIN, M\_ACK, NID\_LRBG + NID\_EM |
| 24 | General message | RBC, RIU | T\_TRAIN, M\_ACK, NID\_LRBG + OP |
| 27 | SH Refused | RBC | T\_TRAIN, M\_ACK, NID\_LRBG + T\_TRAIN of train |
| 28 | SH Authorisation + optional list | RBC | T\_TRAIN, M\_ACK, NID\_LRBG + OP |
| 32 | RBC/RIU System Version | RBC, RIU | T\_TRAIN, M\_ACK, NID\_LRBG + M\_VERSION |
| 33 | MA with Shifted Location Reference | RBC | T\_TRAIN, M\_ACK, NID\_LRBG + OP + D\_REF + 15 |
| 34 | Track Ahead Free Request | RBC | T\_TRAIN, M\_ACK, NID\_LRBG + OP  + D\_REF + D-TAF + L\_TAF |
| 37 | Infill MA | RIU | T\_TRAIN, M\_ACK, NID\_LRBG + OP + 12 + 136 |
| 38 | Initiation of a communication session | RBC | T\_TRAIN, M\_ACK, NID\_LRBG + OP |
| 39 | End of communication session  Acknowledgement | RBC, RIU | T\_TRAIN, M\_ACK, NID\_LRBG + OP |
| 40 | Train Rejected | RBC | T\_TRAIN, M\_ACK, NID\_LRBG + OP |
| 41 | Train Accepted | RBC | T\_TRAIN, M\_ACK, NID\_LRBG + OP |
| 43 | SoM position report confirmed by RBC | RBC | T\_TRAIN, M\_ACK, NID\_LRBG + OP |
| 45 | Assignment of coordinate system | RBC | T\_TRAIN, M\_ACK, NID\_LRBG + OP  + Q\_ORIENTATION |

### Train To Track Radio Message

| **Mess. Id.** | **Message Name** | **Toward** | **Packets / Variables**  ( OP = optional packets) |
| --- | --- | --- | --- |
| 129 | Validated Train Data | RBC | T\_TRAIN, NID\_ENGINE + 0 / 1 + 11 |
| 130 | Request for Shunting | RBC | T\_TRAIN, NID\_ENGINE + 0 / 1 |
| 132 | MA Request | RBC | T\_TRAIN, NID\_ENGINE + 0 / 1  + OP + Q\_MARQSTRESON |
| 136 | Train Position Report | RBC, RIU | T\_TRAIN, NID\_ENGINE + 0 / 1 + OP |
| 137 | Request to shorten MA is granted | RBC | T\_TRAIN, NID\_ENGINE + 0 / 1  + T\_TRAIN of request |
| 138 | Request to shorten MA is rejected | RBC | T\_TRAIN, NID\_ENGINE + 0 / 1  + T\_TRAIN of request |
| 146 | Acknowledgement | RBC, RIU | T\_TRAIN, NID\_ENGINE + 0 / 1  + T\_TRAIN of acknowledged message |
| 147 | Acknowledgement of Emergency Stop | RBC | T\_TRAIN, NID\_ENGINE + 0 / 1  + NID\_EM + Q\_EMERGENCYSTOP |
| 149 | Track Ahead Free Granted | RBC | T\_TRAIN, NID\_ENGINE + 0 / 1 |
| 150 | End of Mission | RBC | T\_TRAIN, NID\_ENGINE + 0 / 1 |
| 153 | Radio infill request | RIU | T\_TRAIN, NID\_ENGINE + 0 / 1  + NID\_C + NID\_BG + Q\_INFILL |
| 154 | No compatible version supported | RBC, RIU | T\_TRAIN, NID\_ENGINE |
| 155 | Initiation of a communication session | RBC, RIU | T\_TRAIN, NID\_ENGINE |
| 156 | Termination of communication session | RBC, RIU | T\_TRAIN, NID\_ENGINE |
| 157 | SoM Position Report | RBC | T\_TRAIN, NID\_ENGINE + 0 / 1 + Q\_STATUS |
| 158 | Text message acknowledged by driver | RBC | T\_TRAIN, NID\_ENGINE + 0 / 1  + NID\_TEXTMESSAGE |
| 159 | Session Established | RBC, RIU | T\_TRAIN, NID\_ENGINE + OP |

## System Data

See subset-026 chap3.18.

### Fixed Values

Data are defined in basic software before a compilation.

### National Values

Data are transmitted with “Country Number” through packet 3 :

* by balise for all levels,
* by radio for level 2/3.

### Train Data

Data must be defined at standstill, before a mission :

1. Train category(ies)
2. Train length
3. Traction / brake parameters
4. Maximum train speed
5. Loading gauge
6. Axle load category
7. Traction system(s) accepted by the engine
8. Train fitted with airtight system
9. List of National Systems available on-board
10. Intentionally deleted
11. Axle number

### Additional Data

Data are provided through DMI :

1. Driver\_ Id
2. ETCS level
3. Radio Network identification
4. RBC Identification
5. Telephone Number
6. ETCS Identity
7. Train Running Number
8. Adhesion factor

### Date and Time

#### UTC

### Data view

TBD.

## Preliminary Data Architecture

As a conclusion of this chapter, we describe a preliminary architecture of stored data with the following objectives :

* To distinguish data associated with speed control and data additional that are involved in secondary function,
* To distinguish linked balise with unlinked balise,
* To facilitate storage of data coming from balise and radio,
* To make easy the computation associated to speed control..

This preliminary description is achieved into several steps in order to facilitate understanding.

### Step 1: set of LRBG :

At this step, we suppose train is meeting only “Linked Balise” and “Linking” is used.

The Database is fill-up as described hereafter :

LRBG\_n-1

LRBG\_n

LRBG\_n+1

D\_link\_1

D\_link\_2

D\_link\_3

Delay

Train\_Position

Window

Comments :

* Balises must be of type “linked” to become LRBG, unlinked balise cannot become LRBG.
* “Linking” is optional : if Linking is not used, LRBG position is given by train position when passing over balise without checking.
* If Linking is used, LRBG position is given by “D\_LINK” of package “linking” and train position is checked and restored accordingly with calculated LRBG position.
* If no Linking is provided, the “Linked Balise” are ignored.
* Condition of acceptance :
  + D\_link + Delay + ½ Window > Train\_Position > D\_link + Delay - ½ Window
  + New Train\_Position = Delay +/- 1 m
  + All packets are stored as they are received.

### Step 2 : set of Unlinked Balise :

At this step, we suppose train is meeting only “Linked Balise” and “Unlinked Balise”.

We suppose “Linking” is not used.

The Database is fill-up as described hereafter :

LRBG\_n-1

BG\_n

BG\_n+1

??

D\_lin

Delay

Train\_Position

Comments :

* Train receives several Unlinked BG in downstream of LRBG\_n+1.
* If “Linking” is not used, the position of Unlinked BG is :
  + D\_link = Train\_Position – Delay
  + All packets are stored as they are received.
  + Packets are stored at LRBG level (because in harmony with radio message).

### Step 3 : set of Packets :

At this step, we deploy all packets that have been stored within each LRBG.

Packets are broken into several “events” that need to be positioned relating to LRBG.

Event\_Position = ( Packet\_Position\_relating\_LRBG ) + ( Event\_Position\_relating\_packet )

We suppose that enough passed LRBG are stored to trake into account all anticipated constraints in downstream of current train position.

The Database is fill-up as described hereafter :

LRBG\_n

Delay

Train\_Position

Event\_n1

Event\_n2

Event\_n3

Comments :

* Event “n1” has been overpassed and cannot be swept, excepted by turning buffer process.
* Event ”n2” & “n3” are events to come and may be swept and replaced every time memorized packets are changing.

### Step 4 : Radio Message :

It is anticipated that Radio Communication behaviour is identical to multiple fictive and ”unlinked” balise

# Processes Over Data

## Data Preliminary Operation

In level 1, data are received by telegrams that are transmitted by balises which are grouped by BG of 1 up to 8 balises.

Each balise telegram is composed of 1023 bits which are composed of :

110 bits of CRC,

83 bits of expansion 11/10 ( 913 -> 830 ),

only 830 bits are used for data.

In case of simple application, telegrams of 341 bits could be involved as follows :

110 bits of CRC,

21 bits of expansion 11/10 ( 231 -> 210 ),

only 210 bits are used for data.

The 830 ( or 210 ) bits of each balise are composed of :

* one header of 50 bits that are used to manage the data of BG and is composed at least of :
  + one balise group identity,
  + the number of balises included in the group,
  + the ranking of each balise,
  + one redundancy indicator.
* “n” packets of variable size to provide mandatory and optional data, each of them including :
  + one packet identity,
  + the length of each packet,
  + the orientation of validity (normal or reverse).
* one “end” packet.

The direction of orientation is given by the order of the balise included in BG :

* Increasing order is “Nominal” orientation,
* Decreasing order is “Reverse” orientation.

Basically, a double track orientation is designed as follows :

Track 1

Nominal

Reverse

LRBG for packets in nominal orientation, Track 1

Track 2

Reverse

Nominal

LRBG for packets in nominal

orientation, Track 2

Reverse

Nominal

LRBG for packets in reverse

orientation, Track 1

Nominal

Reverse

LRBG for packets in reverse orientation, Track 2

BG-1

BG\_3

BG-4

BG\_2

**LRBG Orientation**

Comments :

The drawing hereunder shows a double track with “cross over”, following characteristics hereafter :

* track “1” has its nominal orientation from left to right, by convention,
  + BG\_1 is linked to BG\_3 or further BG to go straight on track\_1,
  + BG\_1 is linked to BG\_2 or further BG to joint track\_2 in reverse orientation.
* track “2” has its nominal orientation from right to left by convention,
  + BG\_2 is linked to BG\_4 or further BG to go straight on track 2,
  + BG\_2 is linked to BG\_1 or further BG to joint track\_1 in reverse orientation.

Notice that any BG can prodive data packets for both orientation, in case of run in both orientation.

But in presence of signal, each BG must be dedicated to one orientation at a time, as hereafter :

Linking normal track 1

Linking reverse track 1

Linking normal track 1

Linking normal track 2

Linking reverse track 2

Linking normal track 2

01

03

05

**02**

**LRBG on a cross-over**

**Comments :**

The drawing hereunder shows a double track with dedicated BG for each signal (recommended) :,

* In normal situation :
  + BG blue are linked toghether in normal orientation and signals are green.
  + BG red are linked toghetherin reverse orientation and signals are red.
* In case of route going from “track\_1” to “track\_2” :
  + BG\_03 is linked to BG\_02,
  + Train can run over track\_2 in reverse direction, once signal\_02 is green.

## Storage of Data

### General Consideration

A Database Architecture needs to be defined in order to classify and to compute data.

All packets must be stored in the database with the following consideration :

* All packets available in nominal orientation (Q\_DIR = nominal) need to be ordered if :
  + Train orientation is nominal (Q\_DIRLRBG = nominal) when passing over LRBG or,
  + If radio track-to-train message refers to a LRBG whose directionality is nominal, and
  + If train is moving ahead (Q\_DIRTRAIN = nominal).
* All packets available in reverse orientation (Q\_DIR = reverse) need to be ordered if :
  + Train orientation is reverse (Q\_DIRLRBG = reverse) when passing over LRBG or,
  + If radio track-to-train message refers to a LRBG whose directionality is reverse, and
  + If train is moving ahead (Q\_DIRTRAIN = nominal).
* If train is moving in reverse (Q\_DIRTRAIN = reverse), no data is stored.
* A maximum of 8 balises must be stored in memory.
* Linking is involved to insure continuity between LRBG and to permit switching between nominal and reverse orientation, in case of double direction of traffic.
* In case of repeat, packet needs to be repeated as much as necessary.
* A Data Ranking is necessary before computation of other functions (MRSP, Integration of brake energy and gravity, etc…).
* A Completeness Checking consists in verifying that no mandatory data is missing : gradient and SSP are defined up to EOA.

### Data Stored on-board

Data are stored or rejected following filters specified in SRS-026 chapter 4.

Accepted data are dispatched following table hereafter with a given destination that could be :

* System Data : Part of Database where are stored parameters.
* Dynamic Database : Database with a turning buffer,
* Train Position : Set of variables related to “Loc-Report”.
* Level and Mode : Set of variables related to Mode and Level,
* TSR Function : Set of variables related to TSR.
* Radio Function : Set of variables related to dialog “trackside / trainborne”.
* Dedicated Function : Set of dedicated variables.

|  |  |
| --- | --- |
| DATA | DESTINATION |
| National Values | System Data |
| Linking | Dynamic Database + Train Position |
| Movement Authority | Dynamic Database |
| Gradient Profile | Dynamic Database |
| International SSP | Dynamic Database |
| Axle load speed profile | Dynamic Database |
| STM max speed | Dynamic Database |
| STM system speed/distance | Dynamic Database |
| Level Transition Order | Level Function |
| Stop Shunting on desk opening | Dedicated Function |
| List of balises for SH area | Dedicated Function |
| MA Request Parameters | Dedicated Function |
| Position Report parameters | Dedicated Function |
| List of Balises in SR Authority + SR mode speed limit and distance | Dedicated Function |
| Temporary Speed Restrictions | TSR Function |
| Inhibition of revocable TSRs from balises in L2/3 | TSR Function |
| Default Gradient for TSR | TSR Function |
| Signalling related Speed Restriction | Dynamic Database |
| Route Suitability Data | Dedicated Function |
| Plain Text Information (location based) | Dedicated Function |
| Plain Text Information (not location based) | Dedicated Function |
| Fixed Text Information (location based) | Dedicated Function |
| Fixed Text Information (not location based) | Dedicated Function |
| Geographical Position | Dedicated Function |
| Mode Profile | Mode Function |
| RBC Transition Order | Dedicated Function |
| Radio Infill Area information | Dedicated Function |
| EOLM information | Dedicated Function |
| Track Conditions excluding big metal masses | Dedicated Function |
| Track condition big metal masses | Dedicated Function |
| Unconditional Emergency Stop | Radio Function |
| Conditional Emergency Stop | Radio Function |
| Train Position | Radio Function |
| Train Data | Radio Function |
| Adhesion factor | Dynamic Database |
| ERTMS/ETCS level | Level Function |
| Table of priority of trackside supported levels | Level Function |
| Driver ID | Radio Function |
| Radio Network ID | Radio Function |
| RBC ID/Phone Number | Radio Function |
| Train Running Number | Radio Function |
| Reversing Area Information | Train Position |
| Reversing Supervision Information | Train Position |
| Track Ahead Free Request | Radio Function |
| Level Crossing information | Dynamic Database |
| Permitted Braking Distance Information | Dynamic Database |
| RBC/RIU System Version | Dedicated Function |
| Operated System Version | Dedicated Function |
| Language used to display information to the driver | Dedicated Function |
| Virtual Balise Covers | Dedicated Function |

## Update of Data

Purge of Data : is automatic by saving the last 8 LRBG.

Save of Data : same as purge, thanks to saving the last 8 LRBG, taking in consideration that a very long train may need 8 LRBG when crossing over a complex station.

## Train Positioning and linking

### Definition of internal variables :

* NID\_LRBG : identifier of reference BG, can be over 24 bits or 14 bits.
* D\_LRBG : distance to reference BG is without sign or orientation.
* Q\_DIR :
  + **qualifier of direction**, attached to most of track-to-train data packet ,
  + must be in compliance with Q\_DIRLRBG to take into account the data,
  + does not exist for train-to-track data packet,
* Q\_DIRLRBG : **qualifier of orientation**,
  + is defined by the order in which balise group is read :
    - 1, 2, 3.. is nominal,
    - 3, 2, 1 is reverse.
  + decoded when overpassing the BG providing the data packets (level 1),
  + or qualifier decoded when taking into account LRBG (level 2/3),
* Q\_DIRTRAIN : **qualifier of running**,
  + determined on-board by cabine number and odometry counter,
  + for instance, nominal is :
    - Front End is cabine1 + counter increasing,
    - or Front End is cabine 2 + counter decreasing,
  + and reverse is :
    - Front End is cabine1 + counter decreasing,
    - or Front End is cabine 2 + counter increasing,
* Q\_DLRBG : qualifier of position,
  + indicate on which side of LRBG is the Front End :
    - should be equal to DIR\_LRBG if no reverse running,
    - but can be changed after a reverse running.
* Q\_LINKORIENTATION : **advance qualifier of orientation,**
  + is given by linking packet.

### Definition of external variables :

The Clock of period 10 ms provides a “Time\_Stamp” for any external variable.

The MMU (odomery) provides the following MMU\_Data at the beginning of each real time cycle :

* Coordinate : 3 absolute counters of distance :
  + C\_estimate : nominal estimated value, so-called “Cn”,
  + C\_doubt-over : maximal value, so-called “Cmax”,
  + C\_doubt-under : minimal value. so-called ”Cmin”.
* Speed : vital speed so-called “Vn”,
* Acceleration : not vital (?) so-callled ”Acc”,
* Motion\_State : vital boolean so-called “Motion”,
* Motion\_Direction : vital boolean so-called “Q\_DIRTRAIN”,
* Time\_Stamp : Timer value “Tn” when evaluating MMU Data ( Cn, Vn, Tn).

The BTM ( balise\_reader) provides 1 up to 8 telegrams composed of :

* Telegram Tel\_k, with k=1..8,
* Time\_Stamp Tk, with k=1..8.

The timing is described hereafter with 4 balises read over 3 real-time cycles :

**B-1**

Tel-1

T1

C1 ?

**B-2**

Tel-2

T2

C2 ?

**B-3**

Tel-3

T3

C3 ?

**B-4**

Tel-4

T4

C4 ?

Cycle\_n

Cn, Vn

Tn

Cycle\_n+1

Cn+1, Vn+1

Tn+1

Cycle\_n+2

Cn+2, Vn+2

Tn+2

**Time**

**Distance**

MMU (odometry) gives Coordinate values with speed and time-stamp at beginning of each real time cycle

BTM (balise reader) gives Telegrams and time stamp when balise axle and antenna axle are alined

Objective is to estimate Coordinate values at each balise axle.

* Cycle “n” : LRBG is positionned by C0,
  + MMU provides Cn, Vn, Tn,
  + BTM has provided Tel-1, T1,
  + Balise B1 position is : C1 = Cn – Vn \* (Tn – T1),
  + Compute C1 with Doubt-Over and Doubt-Under.
* Cycle “n+1” : LRBG is still positionned by C0,
  + MMU provides Cn+1, Vn+1, Tn+1,
  + BTM has provided Tel-2, T2,
  + Balise B2 position is : C2 = Cn+1 – Vn+1 \* (Tn+1 – T2),
  + Compute C2 with Doubt-Over and Doubt-Under.
* Cycle “N+2” : LRBG is still positionned by C0,
  + MMU provides Cn+2, Vn+2, Tn+2,
  + BTM has provided Tel-3, Tel-4, T3, T4,
  + Balise B3 position is : C3 = Cn+2 – Vn+2 \* (Tn+2 – T3).
  + Balise B4 position is : C4 = Cn+2 – Vn+2 \* (Tn+2 – T4),
  + Compute C3 and C4 with Doubt-Over and Doubt-Under.
* IF (Tel-1, Tel-2, Tel-3, Tel-4) are all consistant THEN :
  + Balise B1 becomes LRBG,
  + Reference position is now C1 instead of C0,
  + Order is given to MMU to restore inaccuracy at minimum,
  + Message composed of { Tel-1, Tel-2, Tel-3, Tel-4 } can be stored.

## Emergency Stop

See subset-026 chap3.10. Only in level 2.

## Speed and Distance Monitoring

### Computation of Deceleration and Brake Build up Time

See subset-026 chap3.13.6.

### MRSP computation

See subset-026 chap3.13.7.

### Computation of Target and Curves

See Excel Model on EB curves.

### Supervision Limits Computation

#### Ceiling Speed Monitoring

#### Target Speed Monitoring

#### Release Speed Monitoring

## Protection against undesirable movement

### Roll Away

### Reverse

### Standstill

## Mode Management

See State Machine Diagram of Intercity project.

## Brake command handling

## Special Function

## Version Management

# SysML First Draft

## First Level IBD

Hereafter is a SysML “IBD”first draft of EVC :

Messages

Command / Control

To Position Train

Balise Content (BTM)

Odometry MMU

Train Data

**EVC**

Fixed Values

To Store System Data

To check and Store Balise Content

To Store Track Description in Database & Set of Variables

National Values

To Achieve Processes

DMI Data

To Manage DMI

Radio Message

To check and Store Radio Message

TIU Data

To Manage TIU

Additional Data (changeable on-board)

Orders / Display

Train Position & Speed (Loc\_Report)

Time-Stamp

**Dynamic Database & Set of Variables**

Linking and

re-positionning

System Data

Packets

A1

A8

A2

A3

A4

A5

A6

A7

Packets by radio

**IBD “EVC” of First Level**

Boxes A1, A2, A3 and A4 are dedicated to data management.

Boxe A5 is dedicated to define all processes.

Boxes A6, A7 and A8 are technical function related to devices.

## Second Level

Both boxes ***“To Check and Store Balise Content”*** and ***“To Store Track Description in Database”*** are using the schematic dynamic database matrix as follows :

### General

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| BG\_n | Position | Linking |  |  |  |
|  | Orientat. | Innacur. |  |  |  |
| TrackCond | Pos. | Param. | Asafe | 9,81\*grd | MRSP |
| Others | “ | “ | “ | “ | “ |
| Grade | “ | Value | “ | “ | “ |
| SSP | “ | “ | “ | “ | “ |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| BG\_n+1 | Position | Linking |  |  |  |
|  | Orientat. | Innacur. |  |  |  |
| SSP | Pos. | Value | Asafe | 9,81\*grd | MRSP |
| Others | “ | “ | “ | “ | “ |
| Grade | “ | “ | “ | “ | “ |
| SSP | “ | “ | “ | “ | “ |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| BG\_n+2 | Position | Linking |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| EOA | Pos. | Veoa | Asafe | 9,81\*grd | MRSP |
| DP | “ | 0 | “ | “ | “ |
| OL | “ | 0 | “ | “ | “ |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| BG\_n+3 | Position | Linking |  |  |  |
|  |  |  |  |  |  |
| SSP | Pos. | Value | Asafe | 9,81\*grd | MRSP |
| Others | “ | “ | “ | “ | “ |
| Grade | “ | “ | “ | “ | “ |
| SSP | “ | “ | “ | “ | “ |

xxx

-1%

60 km/h

80 km/h

2%

45 km/h

**Track Layout**

**Type Position Value Asafe Grd Mrsp**

60 km/h

-0,5%

100 km/h

LOA

LRBG

**Dynamic Database of “EVC”**

### Comments

* Assuming the hereunder matrix, the convention is :
  + Direction of travel is top to down, whatever is the direction of travel or orientation,
  + Dimension is enough to keep in memory :
    - * the data related to the last 8 passed balises,
      * and the data related to 8 linked balises as anticipation,
      * only packets in correct orientation are stored.
  + The 3 first columns are filled by packet data :
    - * Type is packet identity,
      * Position is distance from related LRBG,
      * Value is data related to the type,
      * In case of iteration, as much lines as necessary are filled up,
      * Line are classified by increasing position in relation with orientation.
  + The 3 other columns are computed once classification is achieved :
    - * Asafe : emergency brake deceleration, can be a function of speed ) and other data (adhesion factor),
      * Grd : gravitation acceleration or deceleration, depend on related packet (gradient),
      * Mrsp : depends on numerous packet.
  + Other columns can be added for energy computation.
  + Assuming that message received through BG\_n is acting as LRBG :
    - Linking packet is covering 3 BG (BG\_n+1, BG\_n+2, BG\_n+3),
    - MA packet defines an EOA close to BG\_n+2
    - BG\_n+3 is in reverse orientation, meaning a possible change of track,
    - SSP and Gradient packets are enough to cover the track over the EOA (consistency criteria),
    - EOA indicates that route signal is at stop.
    - Assuming later on, when overpassing the BG\_n+1 acting as new LRBG :
  + Linking is verified by Identity and Position regarding the permitted window,

### Management of “Balises Linked” :

It is assumed that “Linked Balises” authorise the following features :

* + - when **any linked balise** is overpassed and consistent, a re-positioning operation is proceeded with cancellation of positioning error (max- min = 1 meter) and this balise becomes the new origine of distance (new LRBG) for further computations.
    - “Linking Information” may be used or not. If not used, (not present), no checking of position is achieved when error is cancelled.
    - **passed information** remained unchanged in order to take them into account as long as necessary,
    - in level 1, **downstream information** are refreshed as soon as new LRBG is set in place.
    - in level 2, **downstream information** remained unchanged up to radio response with a new track description related to new LRBG.
    - in case of TSR, information packet need to be set again as long as no revocation has ocurred.

### Management of “Balises Unlinked” :

It is assumed that “Unlinked Balises” does not authorise the following features :

Balise groups, which are marked as unlinked, shall never be used as LRBG

Justification: The location of an unlinked balise group, or the balise group itself, may not be known to the RBC

## To Position Train

To see what is done by other partners…

A proposed draft is given hereafter :

To Position Train A.8

Train Position & Speed

Synchro Odo / Balise

Linking and

re-positionning

**Linking (input) :**

Id of next BG

Distance to next BG

Orientation of next BG

(Q\_LINKORIENTATION)

Inaccuracy of next BG

**Positioning (output) :**

Id of last BG (NID\_LRBG)

Distance since LRBG (D\_LRBG)

Orientation of LR BG (Q\_DIRLRBG)

Train Position related LRBG (Q\_DLRBG)

L\_DOUBTOVER

L\_DOUBTUNDER

V\_TRAIN

Train Running related LRBG

(Q\_DIRTRAIN)

To Up-date when overpassing linked balise

To Up-date when overpassing re-positioning balise

To Up-date at every real time cycle

To Position when overpassing balise

**Re-Positioning (input) :**

Id of BG

Distance to next BG

Orientation of next BG

**Odometry (Input) :**

Counter Increasing

Counter decreasing

Counter standstill

Counter value

Speed (V\_TRAIN)

**Positioning (input) :**

Id of last BG

Distance since last BG

Orientation of last BG

**“To Position Train”**

Comments :

* To Up-date at every **real time cycle** :
  + Input are:
    - Counter of odometry including :
      * Value, Increasing, Decreasing, Standstill,
      * Speed.
  + Output are up-dated by :
    - D\_LRBG = ( Counter – Counter0 ) \* Qualibration
      * Counter is the current counter of odometry,
      * Counter0 is the counter value when over-passing LRBG.
    - NID\_LRBG = unchanged,
    - Q\_DIRLRBG = unchanged,
    - Q\_DLRBG = inverted if D\_LRBG changes of sign,
    - Q\_DIRTRAIN = inverted if D\_LRBG changes of variation sign,
    - L\_DOUBTOVER = LRBG\_Inaccuracy + ( | D\_LRBG | \* 5% ),
    - L\_DOUBTUNDER = LRBG\_Inaccuracy - ( | D\_LRBG | \* 5% ),
    - V\_TRAIN = Speed + 5%
* To Position when overpassing balise :
  + If balise is **“Re-positioning”,** only D\_LRBG is updated by :
    - D\_LRBG = D\_LINK – L\_SECTION
  + If balise is **linked** with previous one, checking with parameters in red must be true :
    - | ( D\_LRBG – L\_DOUBTOVER < D\_LINK ,
    - | ( D\_LRBG + L\_DOUBTUNDER > D\_LINK ),
    - NID\_LRBG = NID\_BG,
    - Q\_DIRLRBG = Q\_LINKORIENTATION.
  + Output are estimated by :
    - Counter0 = Counter,
    - D\_LRBG = 0,
    - NID\_LRBG = new,
    - Q\_DIRLRBG = new,
    - Q\_DLRBG = inverted if D\_LRBG changes of sign,
    - Q\_DIRTRAIN = Q\_DIRLRBG,
    - L\_DOUBTOVER = LRBG\_Inaccuracy,
    - L\_DOUBTUNDER = LRBG\_Inaccuracy ,
    - V\_TRAIN = Speed + 5%

## To Achieve Processes

See SyML IBD draft hereafter.

SRS-026 3-13-14-1

SRS-026 3-12 & 3-15

SRS-026 3-12-4 & SRS-026 4

SRS-026 3-13-14-2

Command / Control

To Achieve Processes A.5

Orders / Display

Messages

Train Position

& Speed

System Data

Classify DataBase

& MRSP Computation

Asafe

Computation

Supervision Limits Computation

Target and Curves Computation

**Database**

Mode and Level

Protection

Brake Handling

Version Management

Special & Radio

Functions

Curves & Target

MRSP (x)

Asafe (x, v )

SvL

& over speed

& overide

Mode request

& EB request

Mode & Level

Defaut

SRS-026 3-13-3

SRS-026 3-13-8

SRS-026 3-13-9 & 3-13-10

SRS-026 3-13-7

Orders / Display

SRS-026 3-13-17

A5.2

A5.3

A5.1

A5.5 & 6

A5.4

A5.7

**“To Achieve EVC Process”**

The breakdown structure makes to appear a great number of boxes, and each box needs to be analysed through SysML IBD ( Mainly Special and Radio Functions).

The great box in blue corresponds to “SRS-026 3-13” which is so-called “Speed and Distance Monitoring”. The breakdown structure appears within the IBD.

The small box in blue corresponds to “SRS-026 3-12” and to “SRS-026 3-15” which is so-called “Special Functions” ( 1 : mandatory, 2 : optional ).

## Breakdown Structure of EVC

### General

### First Level

The breakdown structure of IBD “first level” could be established as hereafter.

* A1 : To Store System Data
* A2 : To Check and Store Balise Data
* A3 : To Check and Store Radio Messages
* A4 : To Store Track Description in Database
* A5 : To Achieve Processes
* A6 : To Manage DMI
* A7 : To Manage TIU
* A8 : To Position Train

### Second Level

The final breakdown structure of IBD “second level” could be established as hereafter, by distinguishing mandatory and secondary functions.

**Mandatory :**

* A5.1 : Speed and Distance Monitoring :
  + A5.1.1 : Asafe( x, v ) computation,
  + A5.1.2 : MRSP(x ) computation,
  + A5.1.3 : Target and Curves computation,
  + A5.1.4 : Supervision Limits computation,
  + A5.1.5 : Commands.
* A5.2 : Protection :
  + A5.2.1: Emergency Stop (UES, CES, revocation, inhibition)
  + A5.2.2 : Track ahead Free,
  + A5.2.3 : MA Shorten,
  + A5.2.4 : Roll Away Protection,
  + A5.2.5 : Reverse Movement Protection,
  + A5.2.6 : Standstill Supervision.
* A5.3 : Mode and Level Monitoring :
  + A5.3.1: Level Handling,
  + A5.3.2 : Mode Handling.
* A5.4 : Brake Command Handling.

**Secondary :**

* A5.5 : Special Functions 1 :
  + A.5.5.1 : Track Conditions Handling,
  + A.5.5.2 : Route Suitability,
  + A.5.5.3 : Text Transmission,
  + A.5.5.4 : Level Crossing.
* A5.6 : Special Functions 2 :
  + A.5.6.1 : RBC Handover,
  + A.5.6.2 : Non Leading Handling,
  + A.5.6.3 : Splitting/ joining,
  + A.5.6.4 : Reversing Movement,
  + A.5.6.5 : National Systems Handling,
  + A.5.6.6 : Tolerance of Big Metal Mass,
  + A.5.6.7 : Virtual Balise,
  + A.5.6.8 : Route Advanced Display.
* A.5.7 : Version Management.

# Development of A5.1 ( example)

## references

UNISIG Subset\_026 version\_3.3.0

Chapter 3 : ERTMS / ETCS Principes

Chapter 4 : ERTMS / ETCS Modes

Chapter 5 : ERTMS / ETCS Procedures

Chapter 7 : ERTMS / ETCS Language

## overview

[SRS-026-chapter : 3.13.9.3]

“Speed and Distance Monitoring” (A5.1) consists in computation of EBD, EBI, SBD and SBI curves.

“GUI curve” is not addressed.

The use of an Excel database is involved in this exercise.

The technics to fill up the database starting from balise contents is not approached.

The structure of the database is defined and Excel capabilities such as classification of data lists following increasing or decreasing abscissa are used.

We shall start from a typical BG of 2 or 3 balises that are crossed by one train in level 1.

This BG will provide basic data packets such as :

1. National values,
2. Movement Authorization,
3. Balise linking,
4. Speed Profile,
5. Gradient Profile,
6. Level Order,
7. Temporary Speed Restriction.
8. etc…

## Context of Speed and Distance Monitoring

The context is defined with SysML diagram through one BDD (Block Diagram Definition) and several IBD (Internal Block Diagram) that will need to be linked with other parts of the OpenETCS definition.

The Block Diagram Definition should be composed of functions related to chapter 13 of the subset-026 chapter 3.

The Speed and Distance Monitoring ( Function A5.1) is one IBD of BDD :

Speed and Distance

Monitoring A5.1

**(SRS-026 chap3-13)**

Train Length

National Values

SSP Profile

Gradient Profile

Temporary Speed Restriction

Track Condition Request

Speed & Position & LRBG

LOA, Veoa, DP, OL

Mode Profile

Emergency Brake Curves

Service Brake Curves, S, W, P, I

Overspeeding

Fixed Values

Overiding

Target

Request Mode

**Block A51 “Speed and Distance Monitoring”**

Inputs :

* From “Balise Decoding” :
  + - National Values,
    - SSP Profile,
    - Gradient Profile,
    - Temporary Speed Restriction,
    - LOA, Veoa, Danger Point, Overlap.
* From “Internal” :
  + - Fixed Values,
    - Train Length,
    - Request Mode,
  + Track Condition Request.

Outputs :

* Emergency Brake Curves,
* Service Brake Curves (Service, Warning, Permitted, Indication),
* Overspeeding (cell and target),
* Overiding,
* Target,
* Release Speed.

Orders / Display

Train Position

& Speed

System Data

Classify DataBase

& MRSP Computation

Asafe

Computation

Supervision Limits Computation

Target and Curves Computation

**Database**

Curves & Target

MRSP (x)

Asafe (x, v )

SvL

& over speed

& overide

SRS-026 3-13-3

SRS-026 3-13-8

SRS-026 3-13-9 & 3-13-10

SRS-026 3-13-7

Orders / Display

A5.1

**Detail of Block A51 “Speed and Distance Monitoring”**

## Example of Balise Group in level 1

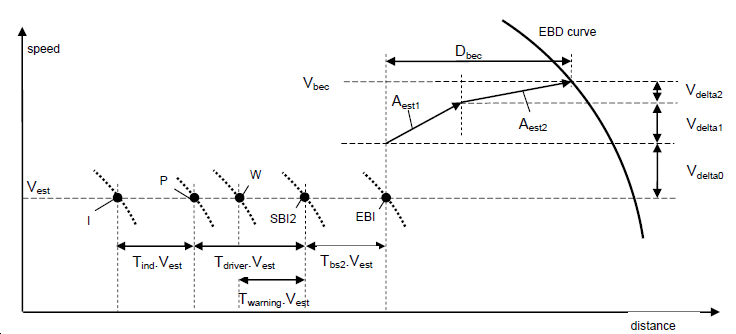
We shall show an example of database configuration for a Balise Group in level 1.

* In first, we shall deploy the database following the singular points that are involved within the BG. We shall see that is not fully meeting the requirement of an EB deceleration variable with the speed.
* In second, we shall deploy the database by quantum of 10 meters.

Definition file of such BG is given within excel file, chapter 5.7.

## EB Supervision Curve

All parameters for braking to target supervision limits for EBD curve are defined within the drawing hereafter :



Braking :

xb, vb, hb

Origin :

,x0, v0, h0

Target :

x1, v1, h1

## EB Supervision Computation

It is defined 3 particular locations :

* Origin :
  + x0 : location where the EB intervenes,
  + v0 : speed at location x0, so-called “Vest”, can be increased by inaccuracy,
  + h0 : hight of gravity centre at x0.
* Braking :
  + xb : location where the EB is active after two phases of transition,
  + vb : speed at location xb, is v0 increased after 2 phases of transition,
  + hb : hight of gravity centre at xb.
* Target :
  + x1 : target location,
  + v1 : target speedn
  + h1 : hight of gravity centre at target.

The next computation will be done through physical mechanic theory in “Energy” by mass unit (m²/s²) between origin and target. To define the EB supervision limits, we put the following equation :

(Brake\_Energy) + (Potential\_Variation\_Energy) > (Kinetic Variation Energy) + (Response\_Correction)

**Cumul\_EB > Cumul\_K**

With the following development :

* + Kinetic Variation Energy = ½ (v0² - v1²)
    - this value concerns the total mass : M \* (1+alfa)
    - alfa can be a function of position “x”.
  + Brake Energy = (x1 – x0) \* Aeb
    - this value concerns the total mass : M \* (1+alfa)
    - Aeb is the average value of the function Asafe(x,v)
    - Asafe(x,v) needs to be integrated between x0 and x1
    - what about v ? (speed at origin x0 or something else )
  + Response\_Correction = ½ [ (Aest1\* T1)² + (Aest2\*T2)²] +(Debc \* Aeb)
    - response is in 2 phases, lasting T1 and T2,
    - the running distance is so-called Debc,
    - brake energy over Debc must be added to kinetic energy
    - power energy (Aest1 and Aest2) must be added too.

* + Potential \_Variation\_Energy = (h1 – h0) \* g / Alpha
    - this value concerns only the mass M,
    - g = 9,81 m/s² must be taken into account,
    - potential energy must be compensated by division with Alpha = 1+alpha,
    - h1 – h0 is the integration of function Grade(x) between x0 and x1.

## EB Supervision Excel File

The use of an Excel database is involved in this exercise in 3 steps :

* + Step 1 : to acquire data from balise content,
  + Step 2 : to classify acquired data into another file,
  + Step 3 : to simul.

Step 1 Excel File :

\_\_\_\_\_\_\_\_\_\_\_\_\_\_



**Position** is the distance of singular point from the reference balise B1,

**Type**  is the nature of singular point,

**DOT** is the Direction of Travel.

**Pos. Inc.** is the incremental position to the next singular point.

**Asafe** is the safe deceleration value of the Emergency Braking, related to increment.

**Grade** is the slope value which permits hight computation, related toincrement.

**Target** is the singular point of MRSP or a DP which can be a potential target.

Step 2 : singular point of database are classified following increasing position.

Step 3 : computation of final parameters.



**EB\_Ener** is the Emergency Brake Incremental Energy.

**G\_Ener** is the Gravitation Incremental Energy.

**Cumul\_EB** is the sum of EB and G energy variation from target DP to current position.

**K\_Ener** is the Kinetic Energy variation from target DP to current position.

**Debc** is the running distance during both transition.

**Corr\_E** is the increased transition energy.

**Cumul\_K** is the sum of Kinetic and Transition Increased Energy.

E\_Maxi is the minimum of both “Cumul”.

SSP\_1 60 km/h is intermediate target and DP is the final target.

## EB Supervision per Quantum

We come back on previous equations :

(Brake\_Energy) + (Potential\_Variation\_Energy) > (Kinetic Variation Energy) + (Response\_Correction)

In order to eliminate the response correction, we shall take in consideration the energy between target ”x1” and current location “xb”. That gives :

(Brake\_Energy\_B) + (Potential\_Variation\_Energy\_B) >= (Kinetic Variation Energy\_B)

and vb <= MRSP(xb)

With the following development :

* + Kinetic Variation Energy B = **½ (vb² - v1²)** ( v1 is target, vb is current v )
  + Brake Energy B = **(x1 – xb) \* Aeb** (integration from xb to x1)
  + Potential \_Variation\_Energy\_B =**(h1 – hb) \* g / Alpha** (integration xb – x1)

The process is to define “vb” from the target location (basically the Danger Point DP) up to the balise location. Then we have at the begining :

* + ½ (vb² - v1²) = (x1 – xb) \* Aeb + (h1 – hb) \* g / Alpha or
  + vb² = v1² + 2 \* [(x1 – xb) \* Aeb + (h1 – hb) \* g / Alpha] and
  + vb <= MRSP(xb)

Starting from DP location (or EOA), to define “x1”,

v1 = 0 or Vsig,

h1 =0

Then to compute vb for each “xb” iteration, with a limitation by MRSP.

When previous MRST is higher, we have a new target.

Then the EBI curve can be computed through Debc and V\_delta computation:

V\_delta = Aest1\*T1 + Aest2\*T2

v0 = vb – V\_delta

Debc = ½ ( Aest1 \* T1² + Aest2 \* T2² ) + v0 \* (T1 + T2)

x0 = xb – Debc

*Variables of speed limit*

Pos. xb is the position of control at speed vb, incremental quantum=10 m. **{ m }**

Type is the type of singular point ( “ \* ”means no type).

DOT is up or down.

MRSP is the Most Restrictive Speed Profile ( compile SSP, TSR, Vmax..). **{ km/h }**

Asafe is safe deceleration value, is a function of x and v0. **{ m/s² }**

Formula : **Asafe(n) = SI((v0 < 60); 0,80; 0,85)**

Grade is slope grade, positive in uphill.

Target is the closest target speed, it can be : MRSP reduction, EOA, DP. **{ km/h }**

EB\_Ener is the Emergency Brake Incremental Energy. **{ m²/s² }**

Formula : **EB\_Ener(n) = Asafe(n) \* Quantum**

G\_Ener is the Grade Incremental Energy. **{ m²/s² }**

Formula : **G\_Ener(n) = Grade \* 9,81 \* Quantum**

Cumul\_EB is cumul “EB\_Ener +G\_Ener”, since smallest target energy. **{ m²/s² }**

Formula : **Cumul\_EB(n) = MIN( (Cumul\_EB(n+1) + EB\_Ener + G\_Ener) ; ½ \* (MRSP/3,6)² )**

v1\_carré is 2 times Kinetic energy of smallest target. **{ m²/s² }**

Formula : **v1\_carré = ½ Target²**

vb is max speed to respect smallest target without response time. **{ km/h }**

Formula **: vb = ( (RACINE( 2 \* Cumul\_EB )) \* 3,6 )**

v\_delta is speed variation during the transition phase T1+ T2. **{ km/h }**

Formula : **v\_delta = ((Aest1\*T1) + (Aest2\*T2)) \* 3,6 )**

v0 is max speed to respect smallest target with response time. **{ km/h }**

Formula : **v0 = vb – v\_delta**

Debc is running distance during response time. **{ m }**

Formula : d = ½ ( (Aest1\*T1²) + (Aest2\*T2²) ) + ( Aest1\* T1 \* T2 )

Formula : Debc = d + (v0 \* (T1+T2) / 3,6)

x0 is max position to respect smallest target with response time. **{ m }**

Formula : x0 = xb - Debc

**Track configuration**

Simple Track Configuration is:

Balise : B1, B2, B3.

SSP : 160 km/h at position 0, 60km/h at position 160.

Grade changes at position 0, 200, 500, 610, 710, 820, 920, 1030, 1040, 1050,1080.

EOA at position 1060 with 30 km/h.

DP at position 1070.

**Train Simulation (for model only)**

Pos.xb is geographical milestone quantum per quantum of 10m = Pos(n)

Pos(0) is initiale position = 0 => Pos(n) = Pos(n-1) + 10

Pos(n+1) = Pos( n) +10

Aest is train acceleration => Aest( n ) = f ( v, n )

Vest is train speed at Pos(n) =>

Vest(n+1) = RACINE( (20 \* Aest( n) ) + Vest(n)² )

Tdelta is time to reach Pos(n+1) and Vest(n+1)

Tdelta = (Vest(n+1) – Vest(n) ) / Aest(n)

Tdelta = { RACINE [ (20 \* Aest( n) ) + Vest(n)² ] – Vest(n) } / Aest(n)

## Additional Issue on EB Supervision per Singular Point

This additional issue aims to address one change of EB deceleration value between two singular points named “A” and “B”.

The initial EB deceleration value is AEB0 and is reduced to AEB1 when the train speed is over “Vs”.

The first step of computation consists in finding the location of Xs related to Vs on the curve AEB0.

The second step consists in evaluating the speed Vc at location “C” or “B” in reason of AEB1 lower than AEB0.

Development are to come.

Xc=Xb

Vc= ?

AEB1

ENERc

Xa

Va

AEBa

ENERa

Xs

Vs

Xb

Vb

AEB0

ENERb

AEB0

AEB1

Vs

S

C

B

A

**AEB change between 2 singular points**

END OF DOCUMENT