<document classification>

openETCS Types for Defining API Inputs and Outputs (from executable model point of view)

Draft for Discussion. API designed for 3rd iteration

**Summary:**

**Company:** openETCS@DB-Netz AG  
**Authors:** Bernd Hekele  
**Reference:** openETCS Modeling (WP3)  
**Index:** openETCS API  
**Date:** Updated Version: 114.01.2015

<summary>

**Distribution List:** <distribution list>

Table Of Contents

[1. General Project Description ToUpdate](#_TOC0)

[2. Software Architecture ToUpdate](#_TOC1)

[2.1. Project Architecture ToUpdate](#_TOC2)

[3. APITypes Project ToUpdate](#_TOC3)

[3.1. API\_Msg\_Pkg Package ToUpdate](#_TOC4)

[3.1.1. Types ToUpdate](#_TOC5)

[3.2. API\_TIU\_Pkg Package ToUpdate](#_TOC6)

[3.2.1. Types ToUpdate](#_TOC7)

[3.3. TIU\_Types\_Pkg Package ToUpdate](#_TOC8)

[3.3.1. Types ToUpdate](#_TOC9)

[3.3.2. Constants ToUpdate](#_TOC10)

List Of Tables

[Table 1: Public Types of API\_Msg\_Pkg ToUpdate](#_TocTable1)

[Table 2: Public Types of API\_TIU\_Pkg ToUpdate](#_TocTable2)

[Table 3: Public Types of TIU\_Types\_Pkg ToUpdate](#_TocTable3)

[Table 4: Public Constants of TIU\_Types\_Pkg ToUpdate](#_TocTable4)

# General Project Description

This modul provides interface definitions for messages defined via the openETCS API.  
  
- Name: API\_Types.etp  
- Description: openETCS Architecture and Desing Document  
- Copyright DB Netz AG, 2015  
- Licensed under the EUPL V.1.1 ( http://joinup.ec.europa.eu/software/page/eupl/licence-eupl )  
- Gist URL: ---  
- Cryptography: No  
- Athor(s): Bernd Hekele  
  
The use of this software is limited to non-vital applications.   
It has not been developed for vital operation purposes and must not be used for applications which may cause harm to people, physical accidents or financial loss.   
  
THEREFORE, NO LIABILITY WILL BE GIVEN FOR SUCH AND ANY OTHER KIND OF USE.

# Software Architecture

## Project Architecture

This section displays the package hierarchy of projects.

Project [APITypes](#LinkM-APITypes)  
 [API\_Msg\_Pkg](#LinkS-!ed/b88d/1CD8/1D0C/546ca1bf57d3)  
 [API\_TIU\_Pkg](#LinkS-!ed/b824/3EFC/133C/54b531897f46)  
 [TIU\_Types\_Pkg](#LinkS-!ed/10ec5/4587/1C64/548456684f7e)

# APITypes Project

## API\_Msg\_Pkg Package

### Types

Table 1: Public Types of API\_Msg\_Pkg

| Name | Definition | Comments and Information |
| --- | --- | --- |
| API\_addInfo\_T | {listLinking : BG\_Types\_Pkg::LinkedBGs\_T} | **Comments:**  packet information received via telegram **listLinking Comments:**  Linking information received via packet 5. Information is of variable length. |
| API\_Telegram\_T | {present : bool, checkResult : bool, api\_bad\_balise\_received : bool, api\_header : BG\_Types\_Pkg::TelegramHeader\_T, api\_packets : API\_Msg\_Pkg::API\_addInfo\_T, centerOfBalisePosition : BG\_Types\_Pkg::centerOfBalisePosition\_T} | **Comments:**  Telegram as received via the API. This type has to be defined in the openETCS API **present Comments:**  Indicates the presence of new and valid information at the start of the routine **checkResult Comments:**  reports the result of the decoding procedure.  true: successfully decoded  false: errors during decoding. Typically, this is the result of the check of the parameters of the telegram. **api\_bad\_balise\_received Comments:**  inicates, whether a bad balise has been received. In this scenario, dta are not valid. **api\_header Comments:**  Telegram\_Header **api\_packets Comments:**  Packets received with this balise **centerOfBalisePosition Comments:**  actual odometry of where the telegram has been received |

## API\_TIU\_Pkg Package

### Types

Table 2: Public Types of API\_TIU\_Pkg

| Name | Definition | Comments and Information |
| --- | --- | --- |
| TIU\_input\_msg | {valid : bool, info : TIU\_Types\_Pkg::Message\_Train\_Interface\_to\_EVC\_T} | **valid Comments:**  Information is valid (true = present) **info Comments:**  Information for the TIU information. Follows API Functional data dictionary |

## TIU\_Types\_Pkg Package

### Types

Table 3: Public Types of TIU\_Types\_Pkg

| Name | Definition | Comments and Information |
| --- | --- | --- |
| A\_nothing\_to\_resume\_profile\_follow\_T | TIU\_Types\_Pkg::nothing\_to\_resume\_profile\_follow\_T ^32 |  |
| Brake\_command\_T | {m\_servicebrake\_cm : TIU\_Types\_Pkg::M\_brake\_signal\_command\_T, m\_emergencybrake\_cm : TIU\_Types\_Pkg::M\_brake\_signal\_command\_T} |  |
| Brake\_inhibition\_command\_T | {m\_regenerativebrake\_cm : TIU\_Types\_Pkg::M\_brake\_inhibit\_command\_T, m\_eddycurrentbrake\_cm : TIU\_Types\_Pkg::M\_eddy\_current\_brake\_inhibition\_T, m\_magneticshoebrake\_cm : TIU\_Types\_Pkg::M\_brake\_inhibit\_command\_T} |  |
| Brake\_pressure\_value\_T | {valid : bool, pressure : int} | **Comments:**  Brake pressure **pressure Comments:**  Unit is [mBar] |
| Brake\_status\_T | {valid : bool, m\_regenerativebrake\_st : TIU\_Types\_Pkg::M\_brake\_status\_T, m\_eddycurrentbrake\_st : TIU\_Types\_Pkg::M\_brake\_status\_T, m\_magneticshoebrake\_st : TIU\_Types\_Pkg::M\_brake\_status\_T, m\_electropneumaticbrake\_st : TIU\_Types\_Pkg::M\_brake\_status\_T, m\_additionalbrake\_st : TIU\_Types\_Pkg::M\_brake\_status\_T} |  |
| cabActiveStatus\_T | enum {cabNone\_Active, cabA\_Active, cabB\_Active} | **Comments:**  Indicates which cab is activated **cabNone\_Active Comments:**  No CAB selected **cabA\_Active Comments:**  CAB A is activated **cabB\_Active Comments:**  CAB B is activated |
| Change\_of\_allowed\_current\_consumption\_T | {d\_test\_current : TIU\_Types\_Pkg::D\_test\_current\_T, m\_current : TIU\_Types\_Pkg::M\_current\_T} |  |
| D\_test\_current\_T | TIU\_Types\_Pkg::D\_test\_distance\_T |  |
| D\_test\_distance\_T | {now : int, distance : int} |  |
| D\_test\_trackcond\_T | TIU\_Types\_Pkg::D\_test\_distance\_T |  |
| D\_test\_trackinit\_T | TIU\_Types\_Pkg::D\_test\_distance\_T |  |
| D\_test\_traction\_T | TIU\_Types\_Pkg::D\_test\_distance\_T |  |
| L\_test\_trackcond\_T | TIU\_Types\_Pkg::D\_test\_distance\_T |  |
| M\_airtightness\_command\_T | enum {tunnel\_condition\_active, tunnel\_condition\_not\_active} |  |
| M\_brake\_inhibit\_command\_T | enum {inhibit\_brake, do\_not\_inhibit\_brake} |  |
| M\_brake\_signal\_command\_T | enum {apply\_brake, release\_brake} |  |
| M\_brake\_status\_T | enum {is\_active, is\_not\_active} |  |
| M\_cab\_signal\_status\_T | enum {both\_desks\_are\_closed, desk\_A\_is\_open, desk\_B\_is\_open, both\_desks\_are\_open} |  |
| M\_change\_traction\_system\_T | {d\_test\_traction : TIU\_Types\_Pkg::D\_test\_traction\_T, m\_voltage : TIU\_Types\_Pkg::M\_voltage\_T} |  |
| M\_current\_T | {no\_restriction : bool, restriction : int} |  |
| M\_directioncontroller\_signal\_status\_T | enum {direction\_controller\_in\_neutral, direction\_controller\_in\_forward, direction\_controller\_in\_backward} |  |
| M\_eddy\_current\_brake\_inhibition\_T | enum {inhibit\_for\_service\_brake, inhibit\_for\_emergency\_brake, inhibit\_for\_both\_service\_emergency\_brake, do\_not\_inhibit\_for\_service\_brake, do\_not\_inhibit\_for\_emergency\_brake, do\_not\_inhibit\_for\_both\_service\_emergency\_brake} |  |
| M\_Isolation\_status\_T | enum {on\_board\_equipment\_is\_isolated, on\_board\_equipement\_is\_not\_isolated} |  |
| M\_mainpowerswitch\_command\_T | enum {open\_main\_power\_swicth, close\_main\_power\_switch} |  |
| M\_nonleading\_signal\_status\_T | enum {M03\_info\_not\_available, non\_leading\_permitted, non\_leading\_not\_permitted} |  |
| M\_pantograph\_command\_T | enum {lower\_pantograph, raise\_pantograph} |  |
| M\_passiveshunting\_signal\_status\_T | enum {passive\_shunting\_permitted, passive\_shunting\_not\_permitted} |  |
| M\_sleeping\_signal\_status\_T | enum {signal\_active, signal\_not\_active} |  |
| M\_trackcond\_T | enum {non\_stopping\_area, tunnel\_stopping\_area, sound\_horn, powerless\_section\_lower\_pantograph, radio\_hole, air\_tightness, switch\_off\_regenerative\_brake, switch\_off\_eddy\_current\_brake\_for\_service\_brake, switch\_off\_magnetic\_shoe\_brake, powerless\_section\_switch\_off\_main\_power\_switch, switch\_off\_eddy\_current\_brake\_for\_emergency\_brake} |  |
| M\_traction\_cutoff\_command\_T | enum {apply\_traction\_cutoff, release\_traction\_cutoff} |  |
| M\_traction\_signal\_status\_T | enum {traction\_on, traction\_off} |  |
| M\_train\_data\_entry\_type\_T | enum {fixed\_entry\_type, flexible\_entry\_type, switchable\_entry\_type, no\_entry\_type} | **Comments:**  Who needs it? **no\_entry\_type Comments:**  Information is undefined |
| M\_trainintegrity\_signal\_status\_T | enum {train\_is\_not\_integer, train\_is\_integer} |  |
| M\_voltage\_T | {voltage\_type : TIU\_Types\_Pkg::M\_voltage\_types\_T, NID\_ctraction : TIU\_Types\_Pkg::NID\_ctraction\_T} |  |
| M\_voltage\_types\_T | enum {line\_not\_fitted\_with\_any\_traction\_system, ac\_25kV\_50Hz, ac\_15kV\_16\_7Hz, dc\_3kV, dc\_1\_5kV, dc\_600\_750kV} |  |
| Message\_EVC\_to\_Train\_Interface\_T | {isolation\_status : TIU\_Types\_Pkg::M\_Isolation\_status\_T, brake\_command : TIU\_Types\_Pkg::Brake\_command\_T, brake\_inhibition : TIU\_Types\_Pkg::Brake\_inhibition\_command\_T, type\_I\_train\_commands : TIU\_Types\_Pkg::Type\_I\_train\_commands\_T, change\_traction\_system : TIU\_Types\_Pkg::M\_change\_traction\_system\_T, passenger\_door\_control\_info : TIU\_Types\_Pkg::Passenger\_door\_control\_info\_T, change\_of\_allowed\_current\_consumption : TIU\_Types\_Pkg::Change\_of\_allowed\_current\_consumption\_T} |  |
| Message\_Train\_Interface\_to\_EVC\_T | {train\_status : TIU\_Types\_Pkg::Mode\_control\_and\_train\_status\_T, brake\_status : TIU\_Types\_Pkg::Brake\_status\_T, brake\_pressure : TIU\_Types\_Pkg::Brake\_pressure\_value\_T, train\_data\_entry\_type : TIU\_Types\_Pkg::M\_train\_data\_entry\_type\_T, train\_data\_info : TIU\_Types\_Pkg::train\_data\_info\_T, type\_I\_train\_and\_brake\_inhibition : TIU\_Types\_Pkg::Type\_I\_train\_and\_brake\_inhibition\_with\_distance\_commands\_T} | **Comments:**  Definition of packages received from the TIU. Each of the components in this structure reflects atomic information. This information might be valid or not valid. **train\_status Comments:**  Status of the train **brake\_status Comments:**  Status of the bakes **brake\_pressure Comments:**  Current brake pressure **train\_data\_entry\_type Comments:**  ??Usecase?? **train\_data\_info Comments:**  !!Incomplete Definition!! What is the purpose of this information? |
| Mode\_control\_and\_train\_status\_T | {valid : bool, m\_sleeping\_st : TIU\_Types\_Pkg::M\_sleeping\_signal\_status\_T, m\_passiveshunting\_st : TIU\_Types\_Pkg::M\_passiveshunting\_signal\_status\_T, m\_nonleading\_st : TIU\_Types\_Pkg::M\_nonleading\_signal\_status\_T, m\_cab\_st : TIU\_Types\_Pkg::M\_cab\_signal\_status\_T, m\_directioncontroller\_st : TIU\_Types\_Pkg::M\_directioncontroller\_signal\_status\_T, m\_trainintegrity\_st : TIU\_Types\_Pkg::M\_trainintegrity\_signal\_status\_T, m\_traction\_st : TIU\_Types\_Pkg::M\_traction\_signal\_status\_T} | **Comments:**  Assumption: if valid all enumerations in this definition are valid. |
| NID\_ctraction\_T | int |  |
| nothing\_to\_resume\_profile\_follow\_T | {d\_test\_trackcond : TIU\_Types\_Pkg::D\_test\_trackcond\_T, l\_test\_trackcond : TIU\_Types\_Pkg::L\_test\_trackcond\_T, m\_trackcond : TIU\_Types\_Pkg::M\_trackcond\_T} |  |
| Passenger\_door\_control\_info\_T | int |  |
| S\_nothing\_to\_resume\_profile\_follow\_T | {nIter : int, value : TIU\_Types\_Pkg::A\_nothing\_to\_resume\_profile\_follow\_T} |  |
| train\_data\_info\_T | {valid : bool, train\_data : int} | **Comments:**  Definition Missing **train\_data Comments:**  not yet defined“” |
| trainData\_T | {validData : bool, RecExit : bool, cabActiveStatus : TIU\_Types\_Pkg::cabActiveStatus\_T} | **Comments:**  Components describing the Train **validData Comments:**  Train Data are valid (linked to SRS 4.8.4 remark [4] **RecExit Comments:**  SRS 4.8.4 remark [1]: for level 2/3:  only if following the reception of the information “Recognition of Exit from TR mode” with a more recent time stamp; for level 1: rejected **cabActiveStatus Comments:**  Indicates which cab is activated |
| Type\_I\_train\_and\_brake\_inhibition\_with\_distance\_commands\_T | {valid : bool, nothing\_to\_resume\_profile\_follow : TIU\_Types\_Pkg::nothing\_to\_resume\_profile\_follow\_T, empty\_profile\_initial\_state\_to\_be\_resumed : TIU\_Types\_Pkg::D\_test\_trackinit\_T} | **valid Comments:**  Valid Indicator for this information **nothing\_to\_resume\_profile\_follow Comments:**  ??Usecase?? **empty\_profile\_initial\_state\_to\_be\_resumed Comments:**  ??Usecase?? |
| Type\_I\_train\_commands\_T | {m\_pantograph\_cm : TIU\_Types\_Pkg::M\_pantograph\_command\_T, m\_airtightness\_cm : TIU\_Types\_Pkg::M\_airtightness\_command\_T, m\_mainpowerswitch\_cm : TIU\_Types\_Pkg::M\_mainpowerswitch\_command\_T, m\_traction\_cutoff\_cm : TIU\_Types\_Pkg::M\_traction\_cutoff\_command\_T} |  |

### Constants

Table 4: Public Constants of TIU\_Types\_Pkg

| Name | Type | Value | Comments and Information |
| --- | --- | --- | --- |
| emptyTIUInfo | TIU\_Types\_Pkg::Message\_Train\_Interface\_to\_EVC\_T | {train\_status : {valid : false, m\_sleeping\_st : signal\_active, m\_passiveshunting\_st : passive\_shunting\_permitted, m\_nonleading\_st : M03\_info\_not\_available, m\_cab\_st : both\_desks\_are\_closed, m\_directioncontroller\_st : direction\_controller\_in\_neutral, m\_trainintegrity\_st : train\_is\_not\_integer, m\_traction\_st : traction\_on}, brake\_status : {valid : false, m\_regenerativebrake\_st : is\_active, m\_eddycurrentbrake\_st : is\_active, m\_magneticshoebrake\_st : is\_active, m\_electropneumaticbrake\_st : is\_active, m\_additionalbrake\_st : is\_active}, brake\_pressure : {valid : false, pressure : 0}, train\_data\_entry\_type : fixed\_entry\_type, train\_data\_info : {valid : false, train\_data : 0}, type\_I\_train\_and\_brake\_inhibition : {valid : false, nothing\_to\_resume\_profile\_follow : {d\_test\_trackcond : {now : 0, distance : 0}, l\_test\_trackcond : {now : 0, distance : 0}, m\_trackcond : non\_stopping\_area}, empty\_profile\_initial\_state\_to\_be\_resumed : {now : 0, distance : 0}}} |  |

End of document.