



# **Le format des données capteurs envoyées par les nœuds ConnecSenS**

**Spécifications – Version TP**

Révision 8



## 1 Quelques informations d'ordre général

Le format de données mis en place cherche à réduire la taille des données transmises, tout en étant générique pour supporter une grande variété de capteurs.

Les données sont codées en binaire pour plus de compacité, en orientation petit-boutiste (little-endian, soit les bits de poids faible en premier).

Le format proposé n'est pas d'une très grande régularité ou simplicité, en particulier il n'est pas composé d'un motif fixe reproduit à l'identique pour toutes les données, comme c'est le cas par exemple pour le format Cayenne qui a servi de source d'inspiration (<https://http://mydevices.com/cayenne/docs/lora/>). Ce choix résulte de la volonté de réduire autant que possible la taille des données transmises. Ainsi, le nombre d'octets d'un champ peut varier et certaines valeurs peuvent « déborder » de leur cadre (ont une portée globale). Pour être plus clair sur ce dernier point j'évoquerais le cas de l'heure (timestamp). Toutes les mesures de capteurs sont horodatées. Il se peut que plusieurs mesures aient une même date, parce qu'elles proviennent d'un même capteur ou encore parce que nous avons reçu simultanément des mesures de plusieurs capteurs par exemple. La répétition à l'identique de données dans une même trame représente une perte de bande passante. Aussi, dans le format proposé ici, les dates d'horodatage (timestamp) sont valides jusqu'à la fin de la trame de données envoyée, ou jusqu'à ce qu'un nouveau timestamp soit défini. Toutes les valeurs de mesure transmises ont donc un timestamp qui est celui en cours de validité, qu'il ait été défini juste avant elles ou bien au début de la trame.

## 2 Glossaire

Voici une définition des termes essentiels à la compréhension du format de données décrit dans ce document :

- **Nœud** : C'est l'organe de collecte et de transmission des données capteurs. Un nœud peut être connecté à plusieurs capteurs. Il y a de fortes chances pour que votre nœud soit un nœud ConnecSenS.
- **Data Channel** : Est une source de données. Le plus logique est de faire coïncider une source de données avec une source physique, par exemple un capteur ou le nœud. Mais ce n'est pas une obligation non plus, un unique capteur physique pourrait être la source de deux, ou plus, *Data Channel* si cette configuration à un sens pour votre application. Les *Data Channel* sont identifiés par un numéro positif et le numéro 0 est réservé pour le nœud.
- **Data Type** : Est un identifiant de type de données. La liste des identifiants est exhaustive (nous l'étudierons plus loin), mais elle peut être allongée pour y ajouter des types de données non encore identifiés. Permet de savoir si les données qui suivent représentent une température, une position géographique, une pression, et ainsi de suite... Il permet également de connaître la taille des données qui le suivent.

### 3 Structure des trames montantes

Il s'agit des trames de données envoyées par le nœud vers la passerelle ou vers le serveur. A l'heure actuelle il n'est pas prévu qu'il y ait des trames descendantes (de la passerelle vers le nœud).

La trame est composée d'un octet de format/version, suivi d'une répétition du motif suivant ( N est toujours  $\geq 1$ ) :

1 octet	1 octet	N octets	N octets	N octets	N octets	...
Format/version	Data Channel n	Data Type X	données	Data Type Y	données	

Où le nombre de couples (*Data Type*, données) est au minimum de 1.

Prenons un exemple pour plus de clarté. Supposons que nous souhaitions envoyer les données de deux capteurs, chacun étant identifié comme une source de données. Le capteur 1 est le *Data Channel 1* tandis que le second capteur est le *Data Channel 2*. Le premier capteur mesure la pression et la température, le second l'humidité. Chacune de ces valeurs est associée à un *Data Type* différent. Appelons *DataTypePress* le *Data Type* qui permet d'identifier une mesure de pression et disons que cette mesure est codée sur deux octets. *DataTypeTemp* correspond à la température codée sur un octet et *DataTypeHumi* à l'humidité codée sur un octet. Nous supposerons également que *DataTypePress* et *DataTypeHumi* sont codés sur un octet tandis que *DataTypeTemp* requière deux octets. Nous obtenons alors la trame suivante :

1 octet	1 octet	1 octet	2 octets	2 octets	1 octet	1 octet	1 octet	1 octet
0xC1	Data Channel 1	DataTypePress	pression	DataTypeTemp	température	Data Channel 2	DataTypeHumi	humidité

Le nombre d'octets utilisés par les *Data Type* et par la valeur de mesure associée est défini par la liste des *Data Type* qui nous étudierons plus loin. De même, le nombre de couples (*Data Type*, données) est déduit du *Data Channel* comme nous le verrons sous peu.

#### 3.1 Le champ de format/version

Il est codé sur un octet et sa valeur est différente de 0x00. Sa structure est la suivante :

<i>Data Channel</i>							
7	6	5	4	3	2	1	0
Format			Version				

La valeur de *Format* est obligatoirement 0x6 (0b110).

La *Version* qui correspond à ce document est la 0x01.

## 3.2 Le champ *Data Channel*

Étudions plus en détails le champ *Data Channel*. Comme indiqué précédemment il permet d'identifier une source de données et de savoir combien de couples (*Data Type*, données) suivront. Il est codé sur un octet dont le découpage est le suivant :

<i>Data Channel</i>							
7	6	5	4	3	2	1	0
ClearGlobal	DataChannelId				NbData		

- `DataChannelId` : est l'identifiant de la source de données à proprement parler. Il est codé sur quatre bits, où le bit 3 de l'octet est celui de poids le plus faible, ce qui autorise 16 identifiants. L'identifiant 0 est réservé pour le nœud, cela nous laisse donc la possibilité d'identifier 15 capteurs.
- `NbData` : indique le nombre de couples (*Data Type*, données) à suivre. Il est codé sur 3 bits, où le bit 0 est celui de poids le plus faible, ce qui laisse donc la possibilité d'avoir jusqu'à 7 couples (la valeur 0 ne peut pas être utilisée et est réservée pour d'éventuels usages futurs).
- `ClearGlobal` : indique si le contexte global doit être vidé ou non. Si ce bit est à 1 alors toutes les valeurs globales courantes n'ont plus cours, nous repartons d'un contexte global vide. Si ce bit est à 0 alors le contexte global courant est toujours d'actualité.

## 3.3 Le champ *Data Type*

Sa longueur minimale est d'un octet et il n'a pas de longueur maximale, même s'il y a peu de chances qu'il dépasse 2 octets un jour. Il est composé d'octets au format identique :

<i>Data Type, octet x</i>							
7	6	5	4	3	2	1	0
More	IdPart						

- `More` : Ce bit (le 7) indique s'il faut un octet supplémentaire pour avoir l'identifiant *Data Type* complet. Ainsi, si l'identifiant est inférieur ou égal à 127, alors un octet, avec le bit `More` à 0 est suffisant. Sinon, le bit `More` est positionné à 1, et un octet supplémentaire contenant les bits 7 à 13 du *Data Type*, et avec `More` à 0, est nécessaire et suit l'octet courant. Et ainsi de suite si deux octets ne sont pas suffisants.
- `IdPart` : Partie de l'identifiant *Data Type*. Souvenez-vous que les données sont transmises en petit-boutisme (les octets de poids faible en premier).

## 3.4 Les différents *Data Type* et les données associées

Certaines valeurs peuvent avoir une portée globale, c'est-à-dire que leur valeur est applicable à tous les capteurs et aux valeurs qu'ils produisent. Ce cas a été illustré dans le premier chapitre avec le cas de l'horodatage des données des capteurs. *A priori*, seul le *Data Channel* 0, celui du nœud, produit des données à valeur globale. La possibilité pour un *Data Type* d'avoir une portée globale est indiquée dans les descriptions à suivre, ainsi que les conditions où cette portée globale est effective.

Voici la liste des identifiants *Data Type* et la taille et le codage des données associées :

### 3.4.1 TimestampUTC (0x00)

TimestampUTC	
Est une valeur d'horodatage (timestamp), exprimée en heure universelle. Est un entier non signé qui compte le nombre de secondes écoulées depuis le 1 <sup>er</sup> janvier 2000 à minuit.	
DataTypeId :	0x00
Portée globale :	Oui, si présent dans le flux du <i>Data Channel 0</i>
Taille <i>Data Type</i> :	1 octet
Taille des données :	4 octets (uint32_t LSB)

### 3.4.2 BattVoltageMV (0x01)

BattVoltageMV	
Est la tension batterie, en millivolts. Codée avec un entier non signé de 16 bits. Peut exprimer une tension batterie comprise entre 0 V et 65,536 V, au millivolt près.	
DataTypeId :	0x01
Portée globale :	Oui, si présent dans le flux du <i>Data Channel 0</i>
Taille <i>Data Type</i> :	1 octet
Taille des données :	2 octets (uint16_t LSB)

### 3.4.3 TempDegC (0x02)

TempDegC

Une température, en degrés Celsius. Codée avec un entier non signé sur 16 bits et une résolution de 0,1 °C par bit sur la plage de -273,1 à 1 365,1 °C. La valeur -273,2 °C, indique une erreur.

Les données sont codées de la façon suivante :

Octet 0	Octet 1							
7-0	7	6	5	4	3	2	1	0
ValueLSB	AlarmH	AlarmL	ValueMSB					

Où :

- La valeur est :  $\text{Octet\_0} \mid ((\text{Octet\_1} \& 0x3F) \ll 8)$
- AlarmL indique si l’alarme de niveau bas est active (1) ou non (0).
- AlarmH indique si l’alarme de niveau haur est active (1) ou non (0).

La température vaut :  $\text{valeur} / 10 - 273,2$ .

DataTypeId :	0x02
Portée globale :	Non
Taille <i>Data Type</i> :	1 octet
Taille des données :	2 octets (uint16_t LSB)

### 3.4.4 TempDegCLowRes (0x03)

TempDegCLowRes	
Une température, en degrés Celsius, avec une résolution et une plage réduite. Codée avec un entier non signé sur 8 bits. Une valeur de 0 correspond à -50 °C et un bit correspond à 0,5 °C. La température maximale codable est donc de 77 °C. Une valeur de 255, soit 77,5 °C, indique une erreur. La température vaut donc : $-50 + \text{valeur} / 2$ .	
DataTypeId :	0x03
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	1 octets (uint8_t)

### 3.4.5 PressAtmoHPa (0x04)

PressAtmoHPa	
Une pression atmosphérique en hectoPascals. Codée avec un entier non signé de 16 bits. Une valeur 0 correspond à 700 hPa et 1 bit correspond à 0,01 hPa. La pression maximale codable est donc de 1 355,36 hPa. La pression vaut donc : $700 + \text{valeur} / 100$ .	
DataTypeId :	0x04
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	2 octets (uint16_t LSB)

### 3.4.6 AirHumidityRelPercent (0x05)

AirHumidityRelPercent	
Une humidité relative de l'air, en pourcents. Codée avec un entier non signé de 8 bits et une résolution de 0,5 % par bit. La température vaut donc : $\text{valeur} / 2$ .	
DataTypeId :	0x05
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	1 octets (uint8_t)

### 3.4.7 GeographicalPosition3D (0x06)

GeographicalPosition3D	
Une position géographique sur trois dimensions (latitude, longitude, altitude). La latitude et la longitude sont exprimées en degrés décimaux au moyen d'un entier signé sur 24 bits où chaque bit correspond à 0,000025° et avec un offset de 180°. L'altitude est exprimée en mètres au moyen d'un entier non signé sur 24 bits où chaque bit correspond à 0,01 mètres et avec un offset de 7 000,00 mètres. Latitude : valeur x 0,000025 – 180,0 Longitude : valeur x 0,000025 – 180,0 Altitude : valeur x 0,01 – 7000,0	
DataTypeId :	0x06
Portée globale :	Oui, si présent dans le flux du <i>Data Channel 0</i>
Taille <i>Data Type</i> :	1 octet
Taille des données :	9 octets : latitude (uint24_t LSB), longitude (uint24_t LSB), altitude (uint24_t LSB)

### 3.4.8 GeographicalPosition2D (0x07)

GeographicalPosition2D	
Une position géographique sur deux dimensions (latitude, longitude). Les valeurs sont exprimées en degrés décimaux aux moyens d'entiers non signés de 24 bits où chaque bit correspond à 0,000025° et avec un offset de 180°. Latitude : valeur x 0,000025 – 180,0 Longitude : valeur x 0,000025 – 180,0	
DataTypeId :	0x07
Portée globale :	Oui, si présent dans le flux du <i>Data Channel 0</i>
Taille <i>Data Type</i> :	1 octet
Taille des données :	6 octets : latitude (uint24_t LSB), longitude (uint24_t LSB)

### 3.4.9 IlluminanceLux (0x08)

IlluminanceLux	
Une luminosité, en Lux. Codé avec un entier non signé sur 16 bits où chaque bit correspond à 1 lux. La luminosité vaut donc : valeur.	
DataTypeId :	0x08
Portée globale :	Non
Taille <i>Data Type</i> :	1 octet
Taille des données :	2 octets (uint16_t LSB)

### 3.4.10 IlluminanceLuxHiRes (0x09)

IlluminanceLux	
Une luminosité en haute précision, en Lux. Codé avec un entier non signé sur 32 bits où chaque bit correspond à 0,0001 Lux. La luminosité vaut donc : valeur x 0,0001	
DataTypeId :	0x09
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	4 octets (uint32_t LSB)

### 3.4.11 SolutionConductivitySCm (0x0A)

SolutionConductivitySCm	
La conductivité d'une solution, en S.cm <sup>-1</sup> . Codé avec un entier non signé de 24 bits où chaque bit correspond à 0,05 S.cm <sup>-1</sup> . La conductivité vaut donc : valeur x 0,05.	
DataTypeId :	0x0A
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	3 octets (uint24_t LSB)

### 3.4.12 DigitalInput (0x0B)

DigitalInput							
Indique l'état logique d'une entrée numérique.							
L'octet des données est codé de la façon suivante :							
Octet des données							
7	6	5	4	3	2	1	0
InputId							Value
Où :							
<ul style="list-style-type: none"><li>InputId est l'identifiant de l'entrée numérique. Nous pouvons donc surveiller jusqu'à 128 entrées.</li><li>Value est l'état de l'entrée numérique</li></ul>							
DataTypeId :		0x0B					
Portée globale :		Non					
Taille <i>Data Type</i> :		1 octet					
Taille des données :		1 octet					



### 3.4.13 VoltageV (0x0C)

VoltageV	
Une tension codée au moyen d'un entier signé sur 16 bits, où chaque bit représente 0,01 V. Peut donc représenter une tension comprise entre -327,68 et 327,67 V. La tension : valeur x 0,01.	
DataTypeId :	0x0C
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	2 octets (int16_t LSB)

### 3.4.14 Acceleration3DG (0x0D)

Acceleration3DG	
Une accélération avec ses trois composantes (X ; Y ; Z). Chaque composante est codée au moyen d'un entier signé sur 16 bits où chaque bit correspond à 0,001 G. Composante : valeur x 0,001.	
DataTypeId :	0x0D
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	6 octets : X (int16_t LSB), Y (int16_t LSB), Z (int16_t LSB)

### 3.4.15 RainAmountMM (0x0E)

RainAmountMM															
La quantité de pluie tombée dans un intervalle de temps.															
L'intervalle de temps est le premier codé dans la trame, en secondes, au moyen d'un entier non signé de 24 bits.															
La valeur, avec une précision de 0,1 millimètres, est représentée par un entier non signé codé sur deux octets de la façon suivante :															
Octets de la valeur															
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
ValueLSB								Res	Alrm	ValueMSB					
Où :															
<ul style="list-style-type: none"><li>La valeur vaut : <code>octet_0   ((octet_1 &amp; 0x3F) &lt;&lt; 8)</code></li><li>Alrm, si positionné à 1 alors indique que la valeur est dans la zone d'alarme.</li><li>Res est un bit réservé pour de futurs usages.</li></ul>															
La quantité de pluie : valeur x 0,1.															
DataTypeId :		0x0E													
Portée globale :		Non													
Taille <i>Data Type</i> :		1 octet													
Taille des données :		5 octets : intervalleSec (uint24_t LSB), quantitéPluie (uint16_t LSB)													

### 3.4.16 BattVoltageFlagsMV (0x0F)

#### BattVoltageFlagsMV

Est la tension batterie, en multiples de 10 millivolts, avec des indications d'état.

Peut exprimer une tension batterie comprise entre 0 V et 40,96 V, avec un pas de 10 mV.

Codée avec un entier non signé de 16 bits de la façon suivante :

Octets															
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
ValueLSB								Réservé	Réservé	Réservé	LowBatt	ValueMSB			

Où :

- La valeur vaut :  $\text{Octet\_0} \mid ((\text{octet\_1} \& 0x0F) \ll 8)$
- LowBatt est positionné à 1 si la tension batterie est tombée en dessous de son seuil de niveau bas.
- Les bits 5 à 7 du second octet sont réservés pour de futurs usages.

La tension batterie vaut : valeur x 10 mV.

DataTypeId :	0x0F
Portée globale :	Oui, si présent dans le flux du <i>Data Channel 0</i>
Taille <i>Data Type</i> :	1 octet
Taille des données :	2 octets (uint16_t LSB)

### 3.4.17 SoilMoistureCb (0x10)

#### SoilMoistureCb

Est l'humidité du sol exprimée en centibars, avec une précision de 1 centibar sur la plage 0 à 254 centibars, et à une profondeur donnée, avec une précision de 1 centimètre sur la plage 1 à 511 centimètres.

Ce Data Type peut contenir plusieurs mesures si un capteur mesure l'humidité du sol à plusieurs profondeurs.

Les données sont composées d'une répétition, 1 fois ou plus, du même motif de format de données. Le motif de base (le format de base) se décline en deux versions : une courte et une longue.

La version courte est utilisée lorsque la profondeur de mesure est inférieure ou égale à 63 centimètres et qu'il n'y a pas d'alarme à signaler. Elle est structurée de la façon suivante :

Octet 0								Octet 1							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
More	0	Depth						Value							

Où :

- More indique s'il y a encore une mesure après celle-ci (1) ou bien si elle est la dernière (0).
- Depth est la profondeur de la mesure en centimètres.
- Value est l'humidité en centibars.

Le format long est utilisé dans les autres cas de figure. Il est structuré de la façon suivante :

Octet 0								Octet 1								Octet 2
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7-0
More	1	DepthLSB						AlarmH	AlarmL	Res	Res	Res	DepthMSB			Value

Où :

- More indique s'il y a encore une mesure après celle-ci (1) ou bien si elle est la dernière (0).
- La profondeur Depth est :  $(\text{Octet\_0} \& 0x3F) \mid ((\text{Octet\_1} \& 0x07) \ll 6)$ .
- AlarmH est positionné à 1 pour signaler que la valeur est au-dessus du seuil haut d'alarme.
- AlarmL est positionné à 1 pour signaler que la valeur est en dessous du seuil bas d'alarme.
- Res sont des bits réservés pour de futurs usages.

La profondeur, en centimètres, vaut : Depth. Une valeur de 0 indique une erreur.

L'humidité, en centibars, vaut : Value. Une valeur de 255 indique une erreur.

DataTypeId :	0x10
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	(2 ou 3 octets) x n, avec $n \geq 1$

### 3.4.18 SoilMoistureCbDegCHz (0x11)

#### SoilMoistureCbDegCHz

Est une mesure de l'humidité du sol par un système qui converti cette humidité en fréquence. Ce DataType peut contenir l'humidité du sol en centibars, mais il contient surtout la fréquence mesurée par le capteur et éventuellement la température du sol. L'idée est de transmettre l'information capteur brute, ainsi que les facteurs à prendre en compte (la température), éventuellement accompagnée de la mesure en centibars, de manière à ce qu'il soit possible de recalculer la valeur en centibars dans le futur si le besoin s'en fait sentir, par exemple suite à une meilleure caractérisation du capteur, ou suite à un nouvel étalonnage du capteur. La fréquence retournée a une précision de 1 Hz, la température de 0,5 °C sur la plage -50 à 77 °C, et la profondeur de 1 centimètre sur la plage 1 à 512 centimètres.

Ce Data Type peut contenir plusieurs mesures si un capteur mesure l'humidité du sol à plusieurs profondeurs.

Les données sont composées d'une répétition, 1 fois ou plus, du même motif de format de données. Le format de données se base sur celui du Data Type SoilMoistureCb, auquel il ajoute à la fin de chaque motif les octets de codage de la température du sol et de la fréquence du capteur :

2 ou 3 octets	1 octet	De 1 à 4 octets
	7-0	
SoilMoistureCb	Temp	Freq

La température, en degrés Celsius est :  $-50 + \text{Temp} / 2$ .

Une valeur de 255, soit 77,5 °C, indique qu'aucune mesure de température n'est disponible.

La fréquence est codée au moyen d'un nombre variable d'octets (de 1 à 4) en fonction de la taille de la valeur. La valeur est codée avec l'ordre petit boutisme (little endian), les bits de poids faible en premier. Chaque octet de fréquence est formaté de la façon suivante :

Octet de fréquence							
7	6	5	4	3	2	1	0
More	FreqPartx						

Où :

- More indique s'il y a encore un octet de fréquence après celui-ci (1) ou bien s'il est le dernier (0).
- FreqPartx est l'octet x de la fréquence, avec x dans l'intervalle [0..3].

La fréquence, en Hz, est donc :

```
(Octet_0 & 0x7F) | ((Octet_1 & 0x7F) << 7) | ((Octet_2 & 0x7F) << 14) |
((Octet_3 & 0x7F) << 21)
```

,dans le cas où quatre octets sont utilisés. Lorsque moins d'octets sont utilisés, alors il suffit de ne pas prendre en compte ceux qui sont manquants.

DataTypeId :	0x11
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	(4 à 8 octets) x n, avec $n \geq 1$

### 3.4.19 SolutionConductivityUSCm (0x12)

#### SolutionConductivityUSCm

La conductivité d'une solution, en  $\mu\text{S.cm}^{-1}$ . Codée avec un entier non signé de 24 bits où chaque bit correspond à  $0,1 \mu\text{S.cm}^{-1}$ . Les deux bits de poids le plus fort indiquent la présence au non d'alarmes. Il est donc possible de représenter une conductivité comprise entre 0 et  $419\,430,3 \mu\text{S.cm}^{-1}$ .

Le format des octets est le suivant :

Octet 0	Octet 1	Octet 2							
7-0	7-0	7	6	5	4	3	2	1	0
ValueLSB0	ValueLSB1	AlarmH	AlarmL	ValueMSB					

Où :

- La valeur est :  $\text{Octet\_0} \mid (\text{Octet\_1} \ll 8) \mid ((\text{Octet\_2} \& 0x3F) \ll 16)$  .
- Le bit AlarmH est positionné à 1 si l'alarme haute est activée, il est à 0 sinon.
- Le bit AlarmL est positionné à 1 si l'alarme basse est activée, il est à 0 sinon.

La conductivité vaut donc : valeur x  $0,1 \mu\text{S.cm}^{-1}$ .

DataTypeId :	0x12
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	3 octets (uint24_t LSB)

### 3.4.20 SolutionConductivityMSCm (0x13)

#### SolutionConductivityMSCm

La conductivité d'une solution, en  $\text{mS.cm}^{-1}$ . Codée avec un entier non signé de 24 bits où chaque bit correspond à  $0,1 \text{mS.cm}^{-1}$ . Les deux bits de poids le plus fort indiquent la présence au non d'alarmes. Il est donc possible de représenter une conductivité comprise entre 0 et  $419\,430,3 \text{mS.cm}^{-1}$ .

Le format des octets est le suivant :

Octet 0	Octet 1	Octet 2							
7-0	7-0	7	6	5	4	3	2	1	0
ValueLSB0	ValueLSB1	AlarmH	AlarmL	ValueMSB					

Où :

- La valeur est :  $\text{Octet\_0} \mid (\text{Octet\_1} \ll 8) \mid ((\text{Octet\_2} \& 0x3F) \ll 16)$  .
- Le bit AlarmH est positionné à 1 si l'alarme haute est activée, il est à 0 sinon.
- Le bit AlarmL est positionné à 1 si l'alarme basse est activée, il est à 0 sinon.

La conductivité vaut donc : valeur x  $0,1 \text{mS.cm}^{-1}$ .

DataTypeId :	0x13
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	3 octets (uint24_t LSB)



### 3.4.21 PressurePa (0x14)

PressurePa									
Une pression, en Pascals. Codée avec un entier non signé de 32 bits où chaque bit correspond à 0,1 Pa. Il est possible de représenter une pression comprise entre – 26 843 545,5 et 26 843 545,5 Pascals.									
Le format des octets est le suivant :									
Octet 0	Octet 1	Octet 2	Octet 3						
7-0	7-0	7-0	7	6	5	4	3	2	1 0
ValueLSB0	ValueLSB1	ValueMSB0	AlarmH	AlarmL	Abs	Sign	ValueMSB1		
Où :									
<ul style="list-style-type: none"><li>• La valeur est : <math>\text{Octet\_0} \mid (\text{Octet\_1} \ll 8) \mid (\text{Octet\_2} \ll 16) \mid ((\text{Octet\_3} \&amp; 0x0F) \ll 24)</math> .</li><li>• Le bit AlarmH est positionné à 1 si l’alarme haute est activée, il est à 0 sinon.</li><li>• Le bit AlarmL est positionné à 1 si l’alarme basse est activée, il est à 0 sinon.</li><li>• Le bit Abs indique si la valeur est absolue (1) ou si elle est compensée de la pression atmosphérique (0).</li><li>• Le bit Sign indique si la valeur est négative (1) ou positive (0).</li></ul>									
La pression vaut donc : $\text{Signe} \times \text{valeur} \times 0,1 \text{ Pa}$ .									
DataTypeId :	0x14								
Portée globale :	Non								
Taille Data Type :	1 octet								
Taille des données :	4 octets (uint32_t LSB)								

### 3.4.22 Config (0x15)

#### Config

Permet de communiquer un élément de configuration d'un Data Channel.

Le Data Type est composé d'un identifiant de paramètre de configuration, d'un octet indiquant longueur de la valeur et de la valeur comme une chaîne de caractères. La chaîne n'est pas terminée par un caractère nul. Le nombre d'octets occupés par la chaîne est donc variable, mais ne peut pas dépasser 31 octets. Si la valeur est vide alors `Len` est à 0 et il n'y a pas d'octets après le second octet.

Le format des octets est le suivant :

Octet 0		Octet 1		Octets 2..n
7-6	5-0	7-5	4-0	
Res	Id	Res	Len	Str

Où :

- `Id` est l'identifiant de paramètre. La liste des valeurs possible est donnée plus bas.
- `Len` est la longueur de la valeur, comprise entre 0 et 31. Elle correspond au nombre d'octets à suivre occupés par la valeur `Str`.
- `Str` est la valeur, codée dans une chaîne de caractères. Ne se termine pas par nul.
- `Res` sont des bits réservés pour de futurs usages.

`DataTypeId` : 0x15

Portée globale : Non

Taille *Data Type* : 1 octet

Taille des données : 2 octets minimum, 33 octets au maximum.

Voici la liste des identifiants de paramètres :

- 0x00 : n'est pas une valeur valide.
- 0x01 : Le nom.
- 0x02 : L'identifiant unique.
- 0x03 : Le type.
- 0x04 : Le numéro de version du micrologiciel (firmware).
- 0x05 : Le nom de l'expérience pour laquelle les données sont produites.
- 0x06 : La période de lecture d'un capteur, en secondes. La chaîne peut être convertie en un entier non signé de 32 bits. Elle peut être 0 si le capteur n'est lu que sur interruption.
- 0x07 : La période d'envoi de la configuration, en secondes. La chaîne peut être convertie en un entier non signé de 32 bits. Une valeur de 0 indique que la configuration n'est pas envoyée périodiquement, mais uniquement si un changement de configuration est détecté.
- 0x08 : La période de lecture du GPS, en secondes. La chaîne peut être convertie en un entier non signé de 32 bits. Une valeur de 0 indique que le GPS n'est pas utilisé.



### 3.4.23 WindSpeedDirection (0x16)

#### WindSpeedDirection

Une mesure de la direction et de la vitesse du vent. La direction est en degrés en sens horaire par rapport au vent de Nord, avec une précision de 1 degré. La vitesse est en mètres par secondes, avec une précision de 0,01 m/s.

La direction est comprise entre 0 et 360°, bornes incluses. 0° est retournée lorsqu'il n'y a pas de vent et 360° lorsque le vent vient du Nord. La vitesse est comprise entre 0 et 81,9 m/s, bornes incluses.

Les données sont encodées de la façon suivante :

Octet 0	Octet 1		Octet 2		
7-0	7-1	0	7	6	5-0
DirLSB	SpeedLSB	DirMSB	SpeedIsCorrected	DirIsCorrected	SpeedMSB

Où :

- La valeur de la direction du vent est :  $\text{Octet\_0} | ((\text{Octet\_1} \& 0x01) \ll 8)$ .
- La valeur de la vitesse du vent est :  $((\text{Octet\_1} \& 0xFE) \gg 1) | ((\text{Octet\_2} \& 0x3F) \ll 7)$ .
- SpeedIsCorrected est à 1 si la vitesse a subi une correction, est à 0 sinon.
- DirIsCorrected est à 1 si la direction a subi une correction, est à 0 sinon.

La direction du vent est : valeur, en degrés.

La vitesse du vent est : valeur x 0,01 m/s.

DataTypeId :	0x16
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	3 octets

### 3.4.24 WindSpeedMS (0x17)

WindSpeedMS			
<p>Une mesure de la vitesse du vent, en mètres par secondes, avec une précision de 0,01 m/s. La vitesse comprise entre 0 et 81,9 m/s, bornes incluses.</p> <p>Les données sont encodées de la façon suivante :</p>			
Octet 0	Octet 1		
7-0	7-6	5	4-0
SpeedLSB	Res	SpeedIsCorrected	SpeedMSB
<p>Où :</p> <ul style="list-style-type: none"> <li>Valeur : <math>\text{Octet\_0}   ((\text{Octet\_1} \&amp; 0x1F) \ll 8)</math>.</li> <li>SpeedIsCorrected est à 1 si la vitesse a subi une correction, est à 0 sinon.</li> <li>Res sont des bits réservés pour de futurs usages.</li> </ul> <p>La vitesse du vent est : valeur x 0,01 m/s.</p>			
DataTypeId :	0x17		
Portée globale :	Non		
Taille Data Type :	1 octet		
Taille des données :	2 octets (uint16_t LSB)		

### 3.4.25 WindDirectionDegN (0x18)

WindDirectionDegN			
<p>Une mesure de la direction est en degrés en sens horaire par rapport au vent de Nord, avec une précision de 1 degré. La direction est comprise entre 0 et 360°, bornes incluses. 0° est retournée lorsqu'il n'y a pas de vent et 360° lorsque le vent vient du Nord.</p> <p>Les données sont encodées de la façon suivante :</p>			
Octet 0	Octet 1		
7-0	7-2	1	0
DirLSB	Res	DirIsCorrected	DirMSB
<p>Où :</p> <ul style="list-style-type: none"> <li>Valeur : <math>\text{Octet\_0}   ((\text{Octet\_1} \&amp; 0x01) \ll 8)</math>.</li> <li>DirIsCorrected est à 1 si la direction a subi une correction, est à 0 sinon.</li> <li>Res sont des bits réservés pour de futurs usages.</li> </ul> <p>La direction du vent est : valeur x 1°.</p>			
DataTypeId :	0x18		
Portée globale :	Non		
Taille Data Type :	1 octet		
Taille des données :	2 octets (uint16_t LSB)		

### 3.4.26 SolarIrradianceWM2 (0x19)

SolarIrradianceWM2		
Une mesure de l'irradiation solaire, en Watts par mètres au carré, avec une précision de 0,1 W/m². L'irradiance est comprise entre 0 et 1 638,3 W/m². Les données sont encodées de la façon suivante :		
Octet 0	Octet 1	
7-0	7-6	5-0
IrrLSB	Res	IrrMSB
Où :		
<ul style="list-style-type: none"> <li>Valeur : <math>\text{Octet\_0}   ((\text{Octet\_1} \&amp; 0x3F) \ll 8)</math>.</li> <li>Res sont des bits réservés pour de futurs usages.</li> </ul>		
L'irradiance est : valeur x 0,1 W/m².		
DataTypeId :	0x19	
Portée globale :	Non	
Taille Data Type :	1 octet	
Taille des données :	2 octets (uint16_t LSB)	

### 3.4.27 RadioactivityBqM3 (0x1A)

RadioactivityBqM3

La mesure de la radioactivité, en Becquerels par mètres cube, avec une précision de 1 Bq/m³.

La longueur de ce Data Type est variable en fonction de la taille des données.

La valeur est composée d'un ou plusieurs octets, avec l'octet de poids faible en premier, tous formatés de la façon suivante :

Octet 0	
7	6-0
More	ValuePart

Où :

- ValuePart est une partie de la valeur.
- More indique si encore au moins un octet supplémentaire est nécessaire pour encoder la valeur.

Ainsi, si le niveau de radiation est compris entre 0 et 127 Bq/m³, alors un seul octet est nécessaire. Pour un niveau entre 128 et 16 383 Bq/m³, deux octets sont nécessaires. Et ainsi de suite.

Le niveau de radiation (Bq/m³) est de : Octet\_0 & 0x7F | (Octet\_1 & 0x7F) << 7 | (Octet\_2 & 0x7F) << 14 | ...

DataTypeId :	0x1A
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	Au moins un octet, cinq au maximum (LSB)

### 3.4.28 HeaterStatus (0x1B)

HeaterStatus	
Indique l'état d'un élément chauffant. Cet état est codé sur un octet selon le format suivant :	
Octet 0	
7-1	0
Res	IsON
Où :	
<ul style="list-style-type: none"><li>IsOn indique si l'élément chauffant est allumé (1) ou éteint (0).</li><li>Res sont des bits réservés pour des usages futurs.</li></ul>	
DataTypeId :	0x1B
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	1 octet

### 3.4.29 DepthCm (0x1C)

DepthCm	
Indique une profondeur, en centimètres, avec une précision de un centimètre. Peut exprimer une profondeur comprise entre 0 et 32 767 centimètres. La valeur est codée sur un ou deux octets, en fonction de la valeur de la profondeur. Si la profondeur est inférieure ou égale à 127 centimètres alors elle est codée de la façon suivante :	
Octet 0	
7	6-0
0	Value
Où :	
<ul style="list-style-type: none"><li>Value est la profondeur en centimètres.</li></ul>	
Si la profondeur est strictement supérieure à 127 centimètres, alors la valeur est codée sur deux octets :	
Octet 0	Octet 1
7	7-0
1	ValueMSB
Où la profondeur vaut : ValueLSB   (ValueMSB << 7) centimètres.	
DataTypeId :	0x1C
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	1 ou 2 octets

### 3.4.30 RelativeDielectricPermittivity (0x1D)

DielectricPermittivity	
Indique une permittivité diélectrique relative. La valeur est codée au format flottant IEEE 754 sur quatre octets (type float en C).	
DataTypeId :	0x1D
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	4 octets

### 3.4.31 SoilVolumetricWaterContentPercent (0x1E)

SoilVolumetricWaterContentPercent		
Indique le pourcentage volumique d'eau dans un sol, avec une précision de 0,01 %. La valeur est encodée sur deux octets selon le format suivant :		
Octet 0	Octet 1	
7-0	7-6	5-0
ValueLSB	Res	ValueMSB
Où :		
<ul style="list-style-type: none"><li>La valeur est : <math>\text{Octet\_0} \mid ((\text{Octet\_1} \&amp; 0x3F) \ll 8)</math>.</li><li>Res sont des bits réservés pour de futurs usages.</li></ul>		
Le pourcentage volumique d'eau dans le sol est : $\text{valeur} / 100$ pourcents.		
DataTypeId :	0x1E	
Portée globale :	Non	
Taille Data Type :	1 octet	
Taille des données :	2 octets	

### 3.4.32 TruebnerSMT100Raw (0x1F)

TruebnerSMT100Raw	
Contiens la valeur « raw » retournée par une sonde d'humidité du sol Truebner SMT100. Cette valeur brute est un entier non signé dont la valeur diminue lorsque l'humidité du sol augmente. Elle est codée au moyen d'un entier non signé de 16 bits.	
DataTypeId :	0x1F
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	2 octets (uint16_t)

### 3.4.33 LevelM (0x20)

LevelM	
Indique un niveau, de liquide le plus souvent, en mètres, avec un pas de 1 millimètre. La valeur est codée sur un entier positif de 16 bits. La valeur est donc comprise entre 0 et 65,535 mètres.	
Le niveau vaut : valeur x 0,001 mètres.	
DataTypeId :	0x20
Portée globale :	Non
Taille <i>Data Type</i> :	1 octet
Taille des données :	2 octets (uint16_t LSB)

### 3.4.34 SolutionSpecificConductivityUSCm (0x21)

SolutionSpecificConductivityUSCm									
La conductivité spécifique d’une solution, en $\mu\text{S}.\text{cm}^{-1}$ . Codée avec un entier non signé de 24 bits où chaque bit correspond à $0,1 \mu\text{S}.\text{cm}^{-1}$ . Les deux bits de poids le plus fort indiquent la présence au non d’alarmes. Il est donc possible de représenter une conductivité comprise entre 0 et 419 430,3 $\mu\text{S}.\text{cm}^{-1}$ .									
Le format des octets est le suivant :									
Octet 0	Octet 1	Octet 2							
7-0	7-0	7	6	5	4	3	2	1	0
ValueLSB0	ValueLSB1	AlarmH	AlarmL	ValueMSB					
Où :									
<ul style="list-style-type: none"><li>La valeur est : <math>\text{Octet\_0} \mid (\text{Octet\_1} \ll 8) \mid ((\text{Octet\_2} \&amp; 0x3F) \ll 16)</math> .</li><li>Le bit AlarmH est positionné à 1 si l’alarme haute est activée, il est à 0 sinon.</li><li>Le bit AlarmL est positionné à 1 si l’alarme basse est activée, il est à 0 sinon.</li></ul>									
La conductivité vaut donc : valeur x $0,1 \mu\text{S}.\text{cm}^{-1}$ .									
DataTypeId :	0x21								
Portée globale :	Non								
Taille Data Type :	1 octet								
Taille des données :	3 octets (uint24_t LSB)								

### 3.4.35 SolutionSpecificConductivityMSCm (0x22)

SolutionSpecificConductivityMSCm									
La conductivité spécifique d’une solution, en mS.cm <sup>-1</sup> . Codée avec un entier non signé de 24 bits où chaque bit correspond à 0,1 mS.cm <sup>-1</sup> . Les deux bits de poids le plus fort indiquent la présence au non d’alarmes. Il est donc possible de représenter une conductivité comprise entre 0 et 419 430,3 mS.cm <sup>-1</sup> . Le format des octets est le suivant :									
Octet 0	Octet 1	Octet 2							
7-0	7-0	7	6	5	4	3	2	1	0
ValueLSB0	ValueLSB1	AlarmH	AlarmL	ValueMSB					
Où :									
<ul style="list-style-type: none"><li>La valeur est : Octet_0   (Octet_1 &lt;&lt; 8)   ((Octet_3 &amp; 0x3F) &lt;&lt; 16) .</li><li>Le bit AlarmH est positionné à 1 si l’alarme haute est activée, il est à 0 sinon.</li><li>Le bit AlarmL est positionné à 1 si l’alarme basse est activée, il est à 0 sinon.</li></ul>									
La conductivité vaut donc : valeur x 0,1 mS.cm <sup>-1</sup> .									
DataTypeId :	0x22								
Portée globale :	Non								
Taille Data Type :	1 octet								
Taille des données :	3 octets (uint24_t LSB)								

### 3.4.36 SensorTypeMM3Hash32 (0x23)

SensorTypeMM3Hash32	
La valeur hachée du nom du type de capteur au moyen de l'algorithme Murmur3, codée sur 32 bits. La graine utilisée est 0x434E5353 en petit boutisme.	
DataTypeId :	0x23
Portée globale :	Oui. Portée globale au DataChannel.
Taille Data Type :	1 octet
Taille des données :	4 octets (uint32_t LSB)

### 3.4.37 ConfigMM3Hash32 (0x24)

ConfigMM3Hash32	
La valeur hachée du fichier de configuration au moyen de l'algorithme Murmur3, codée sur 32 bits. La graine utilisée est 0x434E5353 en petit boutisme.	
DataTypeId :	0x24
Portée globale :	Oui.
Taille Data Type :	1 octet
Taille des données :	4 octets (uint32_t LSB)

### 3.4.38 WindAvgSpeedDirection (0x25)

#### WindAvgSpeedDirection

Une mesure moyenne de la direction et de la vitesse du vent. La direction est en degrés en sens horaire par rapport au Nord, avec une précision de 1 degré. La vitesse est en mètres par secondes, avec une précision de 0,01 m/s.

La direction est comprise entre 0 et 359°, bornes incluses. La vitesse est comprise entre 0 et 81,9 m/s, bornes incluses.

Cette mesure est le résultat d'une moyenne glissante, ou non, sur une période de temps donnée. Cette période est comprise entre 1 et 4095 secondes, bornes comprises. Le pas de moyenne glissante est compris entre 0 et 4095 secondes, bornes incluses. Une valeur de 0 pour le pas indique que la moyenne n'est pas glissante, que la valeur moyenne est remise à zéro au début de chaque période de moyennage.

Les données sont encodées de la façon suivante :

Octet 0	Octet 1		Octet 2		
7-0	7-1	0	7	6	5-0
DirLSB	SpeedLSB	DirMSB	SpeedIsCorrected	DirIsCorrected	SpeedMSB
Octet 3		Octet 4		Octet 5	
7-0		7-4		3-0	
AvgPeriodLSB		RollingLSB		AvgPeriodMSB	
				RollingMSB	

Où :

- La valeur de la direction du vent est :  $\text{Octet\_0} | ((\text{Octet\_1} \& 0x01) \ll 8)$ .
- La valeur de la vitesse du vent est :  $((\text{Octet\_1} \& 0xFE) \gg 1) | ((\text{Octet\_2} \& 0x3F) \ll 7)$ .
- SpeedIsCorrected est à 1 si la vitesse a subi une correction, est à 0 sinon.
- DirIsCorrected est à 1 si la direction a subi une correction, est à 0 sinon.
- La valeur de la période de moyennage est :  $(\text{Octet\_3}) | ((\text{Octet\_4} \& 0x0F) \ll 8)$ .
- La valeur du glissement est :  $((\text{Octet\_4} \& 0xF0) \gg 4) | (\text{Octet\_5} \ll 4)$ .

La direction du vent est : valeur, en degrés.

La vitesse du vent est : valeur x 0,01 m/s.

La période de temps sur laquelle la moyenne est calculée est : valeur, en secondes.

Le pas de glissement du calcul de la moyenne est : valeur, en secondes.

DataTypeId :	0x25
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	6 octets



### 3.4.39 WindAvgSpeedMS (0x26)

#### WindAvgSpeedMS

Une mesure moyenne de la vitesse du vent, en mètres par secondes, avec une précision de 0,01 m/s. La vitesse comprise entre 0 et 81,9 m/s, bornes incluses.

Cette mesure est le résultat d'une moyenne glissante, ou non, sur une période de temps donnée. Cette période est comprise entre 1 et 4095 secondes, bornes comprises. Le pas de moyenne glissante est compris entre 0 et 4095 secondes, bornes incluses. Une valeur de 0 pour le pas indique que la moyenne n'est pas glissante, que la valeur moyenne est remise à zéro au début de chaque période de moyennage.

Les données sont encodées de la façon suivante :

Octet 0		Octet 1	
7-0	7-6	5	4-0
SpeedLSB	Res	SpeedIsCorrected	SpeedMSB
Octet 2		Octet 3	
7-0	7-4	3-0	7-0
AvgPeriodLSB	RollingLSB	AvgPeriodMSB	RollingMSB

Où :

- Valeur :  $\text{Octet\_0} | ((\text{Octet\_1} \& 0x1F) \ll 8)$ .
- SpeedIsCorrected est à 1 si la vitesse a subi une correction, est à 0 sinon.
- Res sont des bits réservés pour de futurs usages.
- La valeur de la période de moyennage est :  $(\text{Octet\_2}) | ((\text{Octet\_3} \& 0x0F) \ll 8)$ .
- La valeur du glissement est :  $((\text{Octet\_3} \& 0xF0) \gg 4) | (\text{Octet\_4} \ll 4)$ .

La vitesse du vent est : valeur x 0,01 m/s.

La période de temps sur laquelle la moyenne est calculée est : valeur, en secondes.

Le pas de glissement du calcul de la moyenne est : valeur, en secondes.

DataTypeId :	0x26
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	5 octets

### 3.4.40 WindAvgDirectionDegN (0x27)

#### WindAvgDirectionDegN

Une mesure moyenne de la direction du vent en degrés en sens horaire par rapport au Nord, avec une précision de 1 degré. La direction est comprise entre 0 et 359°, bornes incluses.

Cette mesure est le résultat d'une moyenne glissante, ou non, sur une période de temps donnée. Cette période est comprise entre 1 et 4095 secondes, bornes comprises. Le pas de moyenne glissante est compris entre 0 et 4095 secondes, bornes incluses. Une valeur de 0 pour le pas indique que la moyenne n'est pas glissante, que la valeur moyenne est remise à zéro au début de chaque période de moyennage.

Les données sont encodées de la façon suivante :

Octet 0		Octet 1	
7-0		7-2	1 0
DirLSB		Res	DirIsCorrected DirMSB
Octet 2		Octet 3	
7-0		7-4	3-0 7-0
AvgPeriodLSB		RollingLSB	AvgPeriodMSB RollingMSB

Où :

- Valeur :  $\text{Octet\_0} | ((\text{Octet\_1} \& 0x01) \ll 8)$ .
- DirIsCorrected est à 1 si la direction a subi une correction, est à 0 sinon.
- Res sont des bits réservés pour de futurs usages.
- La valeur de la période de moyennage est :  $(\text{Octet\_2}) | ((\text{Octet\_3} \& 0x0F) \ll 8)$ .
- La valeur du glissement est :  $((\text{Octet\_3} \& 0xF0) \gg 4) | (\text{Octet\_4} \ll 4)$ .

La direction du vent est : valeur x 1°.

La période de temps sur laquelle la moyenne est calculée est : valeur, en secondes.

Le pas de glissement du calcul de la moyenne est : valeur, en secondes.

DataTypeId :	0x27
Portée globale :	Non
Taille <i>Data Type</i> :	1 octet
Taille des données :	5 octets

### 3.4.41 WindGustSpeedDirection (0x28)

#### WindGustSpeedDirection

Une mesure de la direction et de la vitesse des rafales de vent. La direction est en degrés en sens horaire par rapport au Nord, avec une précision de 1 degré. La vitesse est en mètres par secondes, avec une précision de 0,01 m/s.

La direction est comprise entre 0 et 359°, bornes incluses. La vitesse est comprise entre 0 et 81,9 m/s, bornes incluses.

Cette mesure est le résultat d'une moyenne glissante, ou non, sur une période de temps donnée. Cette période est comprise entre 1 et 4095 secondes, bornes comprises. Le pas de moyenne glissante est compris entre 0 et 4095 secondes, bornes incluses. Une valeur de 0 pour le pas indique que la moyenne n'est pas glissante, que la valeur moyenne est remise à zéro au début de chaque période de moyennage.

Les données sont encodées de la façon suivante :

Octet 0	Octet 1		Octet 2		
7-0	7-1	0	7	6	5-0
DirLSB	SpeedLSB	DirMSB	SpeedIsCorrected	DirIsCorrected	SpeedMSB
Octet 3		Octet 4		Octet 5	
7-0		7-4	3-0	7-0	
AvgPeriodLSB		RollingLSB	AvgPeriodMSB	RollingMSB	

Où :

- La valeur de la direction du vent est :  $\text{Octet\_0} | ((\text{Octet\_1} \& 0x01) \ll 8)$ .
- La valeur de la vitesse du vent est :  $((\text{Octet\_1} \& 0xFE) \gg 1) | ((\text{Octet\_2} \& 0x3F) \ll 7)$ .
- SpeedIsCorrected est à 1 si la vitesse a subi une correction, est à 0 sinon.
- DirIsCorrected est à 1 si la direction a subi une correction, est à 0 sinon.
- La valeur de la période de moyennage est :  $(\text{Octet\_3}) | ((\text{Octet\_4} \& 0x0F) \ll 8)$ .
- La valeur du glissement est :  $((\text{Octet\_4} \& 0xF0) \gg 4) | (\text{Octet\_5} \ll 4)$ .

La direction du vent est : valeur, en degrés.

La vitesse du vent est : valeur x 0,01 m/s.

La période de temps sur laquelle la moyenne est calculée est : valeur, en secondes.

Le pas de glissement du calcul de la moyenne est : valeur, en secondes.

DataTypeId :	0x28
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	6 octets

### 3.4.42 WindGustSpeedMS (0x29)

#### WindGustSpeedMS

Une mesure de la vitesse des rafales de vent, en mètres par secondes, avec une précision de 0,01 m/s. La vitesse comprise entre 0 et 81,9 m/s, bornes incluses.

Cette mesure est le résultat d'une moyenne glissante, ou non, sur une période de temps donnée. Cette période est comprise entre 1 et 4095 secondes, bornes comprises. Le pas de moyenne glissante est compris entre 0 et 4095 secondes, bornes incluses. Une valeur de 0 pour le pas indique que la moyenne n'est pas glissante, que la valeur moyenne est remise à zéro au début de chaque période de moyennage.

Les données sont encodées de la façon suivante :

Octet 0		Octet 1	
7-0	7-6	5	4-0
SpeedLSB	Res	SpeedIsCorrected	SpeedMSB
Octet 2		Octet 3	
7-0	7-4	3-0	7-0
AvgPeriodLSB	RollingLSB	AvgPeriodMSB	RollingMSB

Où :

- Valeur :  $\text{Octet\_0} | ((\text{Octet\_1} \& 0x1F) \ll 8)$ .
- SpeedIsCorrected est à 1 si la vitesse a subi une correction, est à 0 sinon.
- Res sont des bits réservés pour de futurs usages.
- La valeur de la période de moyennage est :  $(\text{Octet\_2}) | ((\text{Octet\_3} \& 0x0F) \ll 8)$ .
- La valeur du glissement est :  $((\text{Octet\_3} \& 0xF0) \gg 4) | (\text{Octet\_4} \ll 4)$ .

La vitesse du vent est : valeur x 0,01 m/s.

La période de temps sur laquelle la moyenne est calculée est : valeur, en secondes.

Le pas de glissement du calcul de la moyenne est : valeur, en secondes.

DataTypeId :	0x29
Portée globale :	Non
Taille Data Type :	1 octet
Taille des données :	5 octets

### 3.4.43 WindGustDirectionDegN (0x2A)

#### WindGustDirectionDegN

Une mesure de la direction des rafales de vent en degrés en sens horaire par rapport au Nord, avec une précision de 1 degré. La direction est comprise entre 0 et 359°, bornes incluses.

Cette mesure est le résultat d'une moyenne glissante, ou non, sur une période de temps donnée. Cette période est comprise entre 1 et 4095 secondes, bornes comprises. Le pas de moyenne glissante est compris entre 0 et 4095 secondes, bornes incluses. Une valeur de 0 pour le pas indique que la moyenne n'est pas glissante, que la valeur moyenne est remise à zéro au début de chaque période de moyennage.

Les données sont encodées de la façon suivante :

Octet 0		Octet 1	
7-0		7-2	1 0
DirLSB		Res	DirIsCorrected DirMSB
Octet 2		Octet 3	
7-0	7-4	3-0	7-0
AvgPeriodLSB	RollingLSB	AvgPeriodMSB	RollingMSB

Où :

- Valeur :  $\text{Octet\_0} | ((\text{Octet\_1} \& 0x01) \ll 8)$ .
- DirIsCorrected est à 1 si la direction a subi une correction, est à 0 sinon.
- Res sont des bits réservés pour de futurs usages.
- La valeur de la période de moyennage est :  $(\text{Octet\_2}) | ((\text{Octet\_3} \& 0x0F) \ll 8)$ .
- La valeur du glissement est :  $((\text{Octet\_3} \& 0xF0) \gg 4) | (\text{Octet\_4} \ll 4)$ .

La direction du vent est : valeur x 1°.

La période de temps sur laquelle la moyenne est calculée est : valeur, en secondes.

Le pas de glissement du calcul de la moyenne est : valeur, en secondes.

DataTypeId :	0x2A
Portée globale :	Non
Taille <i>Data Type</i> :	1 octet
Taille des données :	5 octets

### 3.4.44 AppSoftwareVersionMMPHash32 (0x2B)

AppSoftwareVersionMMPHash32				
Indique la version du logiciel applicatif. La version est composée d'une paire : (majeur.mineur.patch, hash 32 bits).				
Le numéro de version majeur est $\leq 31$ , le mineur $\leq 63$ et de patch $\leq 15$ .				
Les données sont encodées de la façon suivante :				
Octet 0		Octet 1		
7-4	3-0	7	6-2	1-0
MinorLSB	Patch	Dev	Major	MinorMSB
Octet 2	Octet 3	Octet 4	Octet 5	
7-0	7-0	7-0	7-0	
hashB0	HashB1	HashB2	hashB4	
Le numéro de majeur est : (Octet_1 & 0x7C) >> 2.				
Le numéro de mineur est : ((Octet_0 & 0xF0) >> 4)   ((Octet_1 & 0x03) << 4).				
Le numéro de patch est : Octet_0 & 0x0F.				
Si c'est une version de développement alors Dev (Octet_1 & 0x80) est à 1, sinon est à zéro (release).				
Le hash : Octet_2   (Octet_3 << 8)     (Octet_4 << 16)     (Octet_5 << 24).				
DataTypeId :	0x2B			
Portée globale :	Oui, si présent dans le flux du <i>Data Channel 0</i>			
Taille <i>Data Type</i> :	1 octet			
Taille des données :	6 octets			

### 3.4.45 ResetSource (0x2C)

ResetSource		
Indique la source, la cause, du dernier redémarrage du nœud.		
La source est encodée au moyen d’un octet :		
Octet 0		
7-4		3-0
Res		Id
Où :		
<ul style="list-style-type: none"><li>Res sont des bits réservés pour de futurs usages.</li><li>Id est l’identifiant de la source du redémarrage. La liste des identifiants possibles est la suivante :</li></ul>		
Id	Nom	Description
0x0		Réservé.
0x1	BOR	Le nœud a été mis sous tension ou son alimentation est sortie des conditions de fonctionnement normales.
0x2	resetPin	Le bouton « reset » du nœud a été activé.
0x3	software	Le redémarrage fait suite à une demande du logiciel.
0x4	softwareError	Le redémarrage fait suite à une demande du logiciel à cause de la détection d’une erreur.
0x5	watchdog	Le chien de garde est en cause.
0x6	independentWatchdog	Le chien de garde indépendant est en cause.
0x7	lowPowerError	La mauvaise utilisation d’un mode basse consommation est en cause.
0x8	optionBytesLoading	A redémarré pour charger les « option bytes » du microcontrôleur.
0x9	firewall	La pare-feu a causé le redémarrage.
0xA–0xF		Réservés.
DataTypeId :		0x2C
Portée globale :		Non
Taille Data Type :		1 octet
Taille des données :		1 octet

### 3.4.46 DeltaTempDegC (0x31)

DeltaTempDegC	
Représente une variation de température, en degrés Celsius. Elle est codée au moyen d'un entier signé de 16 bits et avec une précision de 0,1 °C par bit sur la plage de -3 276,7 à +3 276,7 °C. La valeur -3 276,8 indique une erreur. La valeur est : $\text{Octet\_0} \mid (\text{Octet\_1} \ll 8)$ . Le deltaT est : valeur x 0,1 °C.	
DataTypeId :	0x31
Portée globale :	Non.
Taille Data Type :	1 octet
Taille des données :	2 octets (int16_t LSB)

## 4 Historique des révisions

Rév.	Date	Modifications
2	Janvier 2019	<ul style="list-style-type: none"> <li>• Ajout du Data Type RainAmountMM.</li> <li>• Ajout du champ Format/Version en début de trame.</li> <li>• Ajout du Data Type BattVoltageFlagsMV.</li> <li>• Ajout du Data Type SoilMoistureCb.</li> <li>• Ajout du Data Type SoilMoistureCbDegCHz.</li> <li>• Ajout de la valeur d'erreur pour le Data Type TempDegC LowRes.</li> <li>• Ajout de la valeur d'erreur pour le Data Type TempDegC.</li> <li>• Ajout du Data Type SolutionConductivityUSCm.</li> <li>• Ajout du Data Type SolutionConductivityMSCm.</li> <li>• Ajout du Data Type PressurePa.</li> <li>• Modification du Data Type TempDegC pour qu'il utilise une plage plus réaliste et pour introduire les niveaux d'alarme.</li> <li>• Modification du format de l'octet de Data Channel pour y ajouter le bit ClearGlobal.</li> <li>• Ajout du Data Type WindSpeedDirection.</li> <li>• Ajout du Data Type WindSpeedMS.</li> <li>• Ajout du Data Type WindDirectionDegN.</li> <li>• Ajout du Data Type SolarIrradianceWM2.</li> <li>• Ajout du Data Type RadioactivityBqM3.</li> <li>• Ajout du Data Type HeaterStatus.</li> <li>• Ajout du Data Type DepthCm.</li> <li>• Ajout du Data Type RelativeDielectricPermittivity.</li> <li>• Ajout du Data Type SoilVolumetricWaterContentPercent.</li> <li>• Ajout du Data Type TruebnerSMT100Raw.</li> <li>• Ajout du Data Type LevelM.</li> <li>• Ajout du Data Type SolutionSpecificConductivityUSCm.</li> <li>• Ajout du Data Type SolutionSpecificConductivityMSCm.</li> </ul>



Rév.	Date	Modifications
3	Avril 2019	<ul style="list-style-type: none"> <li>• Ajout du Data Type <code>SensorTypeMM3Hash32</code>.</li> <li>• Ajout du Data Type <code>ConfigMM3Hash32</code>.</li> </ul>
4	Juin 2019	<ul style="list-style-type: none"> <li>• Ajout de l'identifiant de paramètre 0x05 pour le Data Type <code>Config</code> (0x15).</li> </ul>
5	Juillet 2019	<ul style="list-style-type: none"> <li>• Ajout des identifiants de paramètre 0x06, 0x07 et 0x08 pour le Data Type <code>Config</code> (0x15).</li> </ul>
6	Septembre 2019	<ul style="list-style-type: none"> <li>• Changement de la plage de valeur pour la direction du vent de [0..359] à [0..360] pour s'adapter aux conventions dans ce domaine.</li> </ul>
7	Octobre 2019	<ul style="list-style-type: none"> <li>• Ajout des Data Type <code>WindAvgSpeedDirection</code>, <code>WindAvgSpeedMS</code>, <code>WindAvgDirectionDegN</code>, <code>WindGustSpeedDirection</code>, <code>WindGustSpeedMS</code> et <code>WindGustDirectionDegN</code>.</li> <li>• Ajout du Data Type <code>AppSoftwareVersionMM3Hash32</code>.</li> <li>• Ajout du Data Type <code>ResetSource</code>.</li> </ul>
8	Novembre 2019	<ul style="list-style-type: none"> <li>• Ajout du Data Type <code>DeltaTempDegC</code>.</li> </ul>